Quick and accurate Cellular Automata sewer simulator

Rebecca J. Austin, Albert S. Chen, Dragan A. Savić
and Slobodan Djordjević

ABSTRACT

As urbanisation and climate change progress, the frequency of flooding will increase. Each flood event causes damage to infrastructure and the environment. It is thus important to minimise the damage caused, which can be done through planning for events, real-time control of networks and risk management. To perform these actions, many different simulations of network behaviour are required involving complex and computationally expensive model runs. This makes fast (i.e. real-time or repetitive) simulations very difficult to carry out using traditional methods, thus there is a requirement to develop computationally efficient and accurate conceptual sewer simulators. A new Cellular Automata (CA) based sewer model is presented which is both fast and accurate. The CA model is Lagrangian in nature in that it represents the flow as blocks, and movement of the blocks through the system is simulated. To determine the number of blocks which should be moved it uses either the Manning's or Hazen–Williams equation depending on the flow conditions to calculate the permitted discharge. A case study of the sewer network in Keighley, Yorkshire, is carried out showing its performance in comparison to traditional sewer simulators. The benchmarks used to verify the results are SIPSON and SWMM5.

Key words | Cellular Automata, sewer modelling, urban flooding

Rebecca J. Austin (corresponding author)
Albert S. Chen
Dragan A. Savić
Slobodan Djordjević
Centre for Water Systems,
College of Engineering,
Mathematics & Physical Sciences,
University of Exeter,
North Park Road,
Exeter,
EX4 4QF,
UK
E-mail: *rja210@exeter.ac.uk*

NOTATION

A	Cross sectional area (m^2)
B	Water table width (m)
C	Hazen–Williams roughness coefficient (–)
c	Configuration of a grid in CA (–)
D	Pipe diameter (m)
f	Local transition rule (–)
G	Global transition rule (–)
g	Gravitational acceleration ($m\,s^{-2}$)
H	Set of all possible states (–)
h_{DS}	Water level at the DS (m)
h_f	Headloss (m)
h_{US}	Upstream water level (m)
I_D	Downstream invert (m)
I_U	Invert at upstream (m)
k	Conversion factor, for SI $k = 0.849$ (–)
M_1	Manning's equation multiplying coefficient (–)
M_2	Hazen–Williams equation multiplying coefficient (–)
O_1	Model1 output (–)
O_2	Model2 output (–)
N	Number of elements in data set (–)
n	Manning's roughness coefficient (–)
Q	Flow rate ($m^3\,s^{-1}$)
Q_i	Inflow ($m^3\,s^{-1}$)
Q_o	Outflow ($m^3\,s^{-1}$)
R	Hydraulic radius (m)
r	Radius (–)
S	Storage (m^3)
S_f	Friction slope (–)
S_0	Slope of pipe (–)
Δt	Time step length (s)
t	Time (s)
t_T	Arrival time step(s)
V	Velocity ($m\,s^{-1}$)
v	A cell in a grid used by CA (–)

doi: 10.2166/hydro.2014.070

W Block size (m^3)
x Space coordinate (m)
y Water depth (m)
z Arbitrary time step(s)

INTRODUCTION

Numerous storm events occur each year. Some of these events are severe enough to cause flooding. The number resulting in flooding is increasing, which is of particular concern in urban environments due to the cost of the damage that occurs as a consequence. The UK government has estimated that flooding in England and Wales affects 80,000 homes and causes £270 million damage annually (Evans & Office of Science and Technology 2004). The number of properties being affected is expected to rise to between 300,000 and 400,000 properties by 2080. As well as properties, flooding can damage infrastructure and the environment, further increasing the cost.

Flooding is increasing for a number of reasons, including climate change and urban creep (White 2008). It is accepted that climate change is causing the return period of storms to decrease, thus severe storms are occurring at a higher frequency. Urban creep is causing the increase in impermeable surfaces within a catchment, which reduces infiltration and leads to a higher and faster peak runoff. With more frequent and intense flooding around the world, the ability to model sewer flows quickly and accurately is becoming more desirable.

Many sewer models have been developed for use in both industry and research. Currently most sewer models use the Saint Venant Equations (SVE), to simulate the flow

$$\frac{\partial y}{\partial t}+\frac{1}{B}\frac{\partial Q}{\partial x}=0 \tag{1}$$

$$\frac{\partial Q}{\partial t}+\frac{\partial}{\partial x}\left(\frac{Q^2}{A}\right)+gA\frac{\partial y}{\partial x}-gA(S_0-S_f)=0 \tag{2}$$

where Q = flow rate ($m^3\,s^{-1}$); t = time (s); A = cross sectional area (m^2); S_f = friction slope (–); y = water depth (m); g = gravitational acceleration ($m\,s^{-2}$); S_0 = slope of pipe (–); B = water table width (m); and x = space coordinate (m).

This system of equations must be solved using numerical algorithms, making them slow and complex to use (Meirlaen & Vanrolleghem 2000). Attempts like the DORA Algorithm (Noto & Tucciarelli 2001) have been developed to boost the speed of this type of model. The DORA Algorithm splits the Momentum Equation into two parts, which are then solved separately. The calculation time when using the DORA Algorithm was reduced by 20–25% in comparison to using traditional methods. However, the computation cost was still high. For more significant savings, a completely different approach, such as the use of conceptual models, is required. Conceptual models do not solve the SVE but the mass conservation is still satisfied (Achleitner *et al.* 2007) via the Storage Equation

$$\Delta S=\frac{Q_i-Q_o}{\Delta t} \tag{3}$$

where S = storage (m^3); Q_o = Outflow ($m^3\,s^{-1}$); Q_i = Inflow ($m^3\,s^{-1}$); and Δt = time step (s).

A variety of conceptual models have been developed with the first being the Muskingum Method (McCarthy 1938) cited for example by Samani & Jebelifard (2003). This method transforms the Storage Equation into a system of equations which can be solved explicitly. The Muskingum Method has been used many times in the literature to simulate sewers such as in the CITY DRAIN model (Achleitner *et al.* 2007). The biggest drawback of the Muskingum Method is that its level of accuracy depends on two parameters: the storage-time constant and the weighting factor (Singh & McCann 1980). The two parameters are estimated using a number of techniques including, for example, Graphical Methods and Direct Optimisation Methods (Chatila 2003). The calibration of these two parameters is cumbersome which makes the Muskingum Method computationally expensive as multiple runs are required.

Linear reservoir models are another common type of conceptual model. Many have been developed such as KOSIM (ITWH 1995), REMULI (Vaes 1997), and Cosmoss (Calabrò 2001). Reservoir models use Nash–Cascades and a transportation time (Paulsen 1986; ITWH 1995) cited in Meirlaen *et al.* (2000) to simulate the flow. Linear reservoir

models work by moving the flow through either a single reservoir or a sequence of reservoirs in place of either a pipe, group of pipes or entire networks. A retention constant is used to determine how long the flow remains in each reservoir before it moves on to a connected reservoir (Vanrolleghem *et al.* 2009). Most reservoir models do not consider downstream boundary conditions, thus they cannot simulate backwater effects. This causes reservoir models to overestimate the maximum flow rates and peaks, thus having lower accuracy than those using the SVE (Vanrolleghem *et al.* 2009). An exception to the exclusion of backwater effects is KOSIM-WEST (Solvi *et al.* 2008). This model used a combiner-splitter technique to allow reverse flow through the system when flooding occurs. If a manhole is flooded, the excess was 'split' from the flow at the manhole and 'combined' back into the network at the upstream node. A key drawback of KOSIM-WEST, however, is that due to its use of an adaptive time step it is considerably slower than KOSIM (Solvi 2007).

Another conceptual modelling approach is the use of Lagrangian models. These models simulate the movement of the individual (or a group of) particles in the flow. An example of this type of model is FastNett (Fullerton 2004). FastNett uses the Packet Approach (Thomas (2000) cited by Fullerton (2004)) to simulate the flow. The Packet Approach works by taking a volume of flow entering the pipe to be a packet. A packet then moves to the node at the downstream end of the pipe and does not interact with any others while it is in the pipe. This model simulated a network in 1/50th of the time a hydrodynamic model (based on the SVE) took on the same system. However, FastNett was unable to model diverging flow, structures within a sewer system (such as a weir) or backwater effects. This severely restricts the applicability of the model and reduces the accuracy.

The lower accuracy in the conceptual models developed thus far has limited their use. Improving the accuracy is essential to widen the application of fast conceptual models. To obtain a viable alternative to traditional modelling methods, low computational times and high accuracy are crucial. A compromise though must be made between the two as obtaining a high level of accuracy requires a large number of calculations which in turn increases the computation time.

We will describe a fast and accurate conceptual sewer simulator developed using Cellular Automata (CA). This new model will simulate both surcharged and free surface flow within a sewer network. It determines whether the free surface flow is supercritical or subcritical, thus allowing the flow rate to be calculated accurately. CA are dynamical systems able to replicate a real system through the use of simple rules and states (Fuk 2004). CA will be introduced more formally in the following section and their use in similar fields will be discussed briefly.

Following the introduction of CA, the methodology of the new CA sewer simulator will be presented. A case study is carried out by simulating the sewer network in Keighley, Yorkshire (UK). The results of the new model and two recognised hydrodynamic models, SWMM 5 (Rossman 2010) and SIPSON (Djordjevic *et al.* 2005), are compared on the network. Both the simulation of the flow and the level of efficiency are looked at. The root mean square error (RMSE), the normalised root mean square error (NRMSE) and the Nash–Sutcliffe Efficiency (NSE) (Nash & Sutcliffe 1970) of the results are calculated to determine the level of agreement between the models. The strength and weakness of the new model are then discussed, as well as areas of future research.

CELLULAR AUTOMATA

CA represent an alternative approach to conceptual modelling of complex systems. John Von Neumann and Stanislaw Ulam introduced CA in the late 1940s to simulate self-replicating systems in biology (Von Neumann 1966). CA simulate complex dynamical systems using simple transition rules that change the states of the cells, which represent the conditions within the cell. The theory of CA has been developed further and presented in the literature, for example, Wolfram (1982, 1984), Albert & Culik (1987) and Morita (2012). The basic theory will be presented here.

In CA, the region being simulated is made up of cells and is referred to as a lattice or grid. Traditionally lattices are infinite, although finite lattices are commonly used in real world applications. Every cell within the region has a neighbourhood, which includes the cell being simulated and those surrounding it. A neighbourhood is denoted by

N. The neighbourhood affects the simulation of the cell. The layout of a neighbourhood varies between CA models. Traditionally one-dimensional (1D) CA have a radial neighbourhood which can be represented as a simple line of cells (Figure 1). This type of neighbourhood can also be simply described by Equation (4) where r is referred to as the radius (Durand *et al.* 2003)

$$N = \{-r, \ldots, 0, \ldots r\} \tag{4}$$

In the work carried out by Wolfram (1982), a radial neighbourhood was used with a radius of 1. When 1D CA with this neighbourhood has only two states, this can be referred to as elementary CA (Wolfram 1983).

Each cell is assigned a state and the simulation is carried out by updating these states. Traditionally states are discrete and indicate the condition of the cell such as in the famous CA Conway's Game of Life, which is widely known from the work done by Gardner (1970). In this CA, a cell has a state of 1 if it is alive and 0 if it is dead. The states are changed using simple transition rules depending on the state of the cell and its neighbours. More formally CA is a triple $A = (H, N, f)$. In the triple H is the set of all the possible states which can occur in the CA. The neighbourhood is represented within the triple by N. The size N is denoted by n. The final member of the triple is f which is the local transition rules. The local transition rules, Equation (5), change the state of a cell depending on the states assigned to the neighbouring cells. There can be both local and global transition rules. The global transition rule, denoted by G and shown in Equation (6), describes how the complete configuration, c, changes to the next. The difference between the two types of transition rules are explained more clearly in Figure 2

$$f{:}H^n \rightarrow H \tag{5}$$

$$G_f(c)_v = f(c_{v+N}) \tag{6}$$

where v = cell in the lattice (–).

CA has been used to model many different situations and processes, for example, lava flow (Miyamoto & Sasaki 1997), fire evacuation (Yang *et al.* 2002) and evaluation of future scenarios and expansion of urban drainage networks (Sanchez *et al.* 2014). Similar models have also been used in hydrology to model surface flow. Dottori & Todini (2010, 2011) developed a CA model for overland flow modelling. This model divides the area being modelled into cells and then simulates each cell independently using a form of the SVE. Initially the Diffusion Wave was used but to improve the stability of the model a version that used the inertia formulation (Bates *et al.* 2010) was then developed (Bates *et al.* 2010; Dottori & Todini 2011). This model is significantly faster than other surface models which use the SVE; however, it is still relatively complex and an even faster model would be desirable. Recently, Ghimire *et al.* (2012) presented an alternative CA approach for two-dimensional (2D) modelling that uses regular grid cells as a discrete space for the CA setup and applies generic rules to local neighbourhood cells to simulate the spatio-temporal evolution of pluvial flooding. Dottori & Todini (2010,

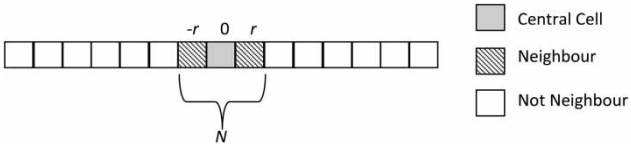

Figure 1 | 1D neighbourhood schematic where radius, r, is 1.

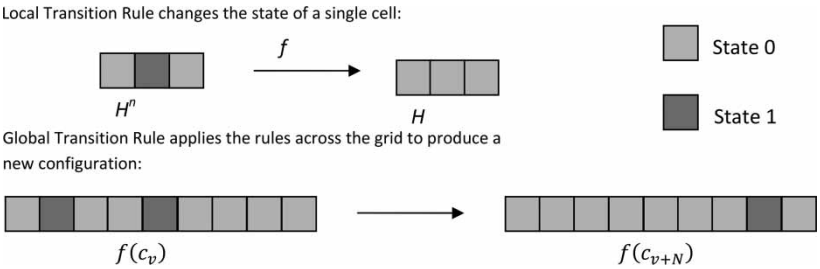

Figure 2 | Global and local transition rules, with $n = 3$ in both.

2011) also applied CA to a limited number of 1D cases. They found that their model could also produce accurate results with a low computation time. However, the 1D case still only modelled surface flow and was not applied to pipe flow where transition to pressurised flow, and vice versa, could occur. The simulation of 1D pipe flow is more complicated and to obtain even faster CA models a fully conceptual model is required. The methodology for such a model is presented in the following section.

METHODOLOGY

The new model is called the Block Cellular Automata 1D Model, BCA1D. It is a CA model that uses a Lagrangian approach to simulate sewer flow. BCA1D represents the flow within the sewer system in the form of figurative water blocks, that is, finite volume of water. These blocks are used to move water from one manhole to another through a connecting pipe, as shown in Figure 3. The number of blocks that can move at each time step depends upon the depths and the water levels within the current and receiving manhole, as well as the characteristics of the connecting pipe. BCA1D is a very simple sewer simulator that uses a small number of steps to carry out a CA simulation of the sewer network. The basic steps of the algorithm are explained in Figure 4.

BCA1D uses the water levels at the upstream and downstream ends of each pipe to calculate the hydraulic gradient and the maximum flow rate. The depth in the upstream manhole indicates if the pipe is surcharged or not which determines the correct equation to use, Equations (8) or (9). The volume allowed to move (the maximum number of blocks) is obtained by multiplying the flow rate by the length of the time step in use. In the example shown in Figure 3, the maximum number of blocks which can leave Manhole 2 is three. The volume obtained is then divided by the block size and the number of blocks allowed to move is obtained. The block size is set prior to the simulation and is chosen small enough that flow will be able to move through the smallest pipe in the network during low flow rates. This gives the basic form of the Transition Rule, as given in Equation (7)

$$T_i = \text{number of blocks} = \frac{Q\Delta t}{W} \tag{7}$$

where W = block size (m^3).

In the literature, numerous different equations have been developed that are capable of calculating the flow rate in pipes. Any of these could be incorporated into this Transition Rule, making the model very flexible. As the model is currently being developed to simulate sewer pipe flow it is important that the network characteristics are respected and the different types of flow which occur in a network can be simulated. To do this there are two forms of this Transition Rule; one to accommodate for pressurised flow and the other to model free surface flow. The flow rate, Q, is a function that changes depending on the depth at the upstream manhole. In the current form of the model, the

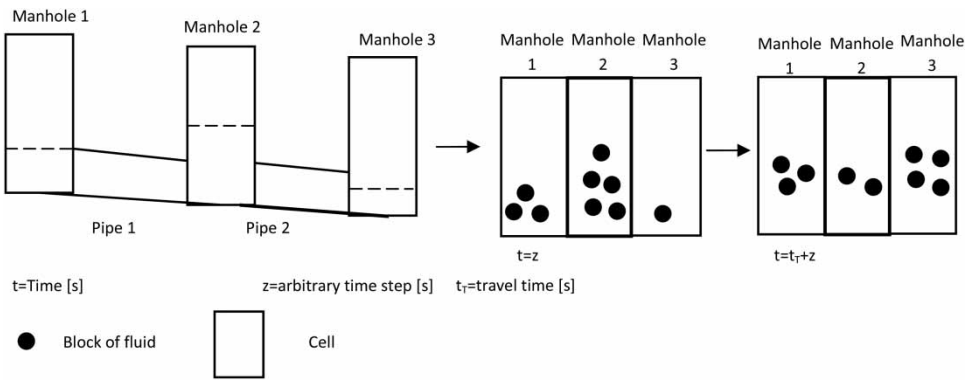

Figure 3 | Schematic description of the simulation process of a manhole, in example Manhole 2 is being simulated.

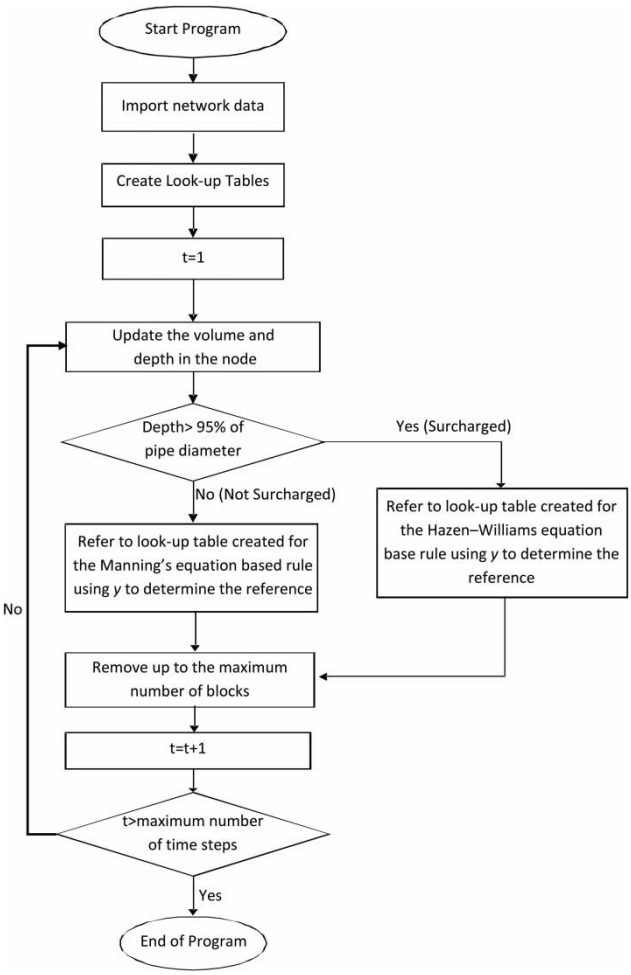

Figure 4 | Flow chart of the algorithm steps.

Manning's equation, Equation (8), is applied if the pipe flow is not surcharged. Otherwise, the Hazen–Williams equation, Equation (9), is used if the pipe is surcharged. These equations have been selected to allow pipe flow to be simulated accurately. This also means that the parameters included in the model only allow for flow in pipes to be simulated, that is, in its current state it is not suitable for the simulation of open channel flow such as in rivers or streams

$$Q = \frac{1}{n} A R^{2/3} \sqrt{S_f} \tag{8}$$

$$Q = kCAR^{0.63} S_f^{0.54} \tag{9}$$

where n = Manning's roughness' coefficient (–); C = Hazen Williams roughness coefficient (–); k = conversion factor, for SI $k = 0.849$ (–); R = hydraulic radius (m).

The use of the Hazen–Williams and Manning's equations allows the model to simulate both free surface and surcharged flow conditions. The model can automatically switch from one equation to the other based on flow condition. Nevertheless, the details of flow transition are not considered as our main focus is to simulate a sewer network throughout a storm event.

By combing these equations with the basic form of the Transition Rule, the two rules used to carry out the simulation are obtained, Equations (10) and (11). The values of variable parameters depend upon the water depth at the upstream and downstream ends of the pipe. Making the depth the dependent variable, it can be taken to be the state of the cells

$$T_1 = \frac{A\Delta t}{Wn} R^{2/3} \sqrt{S_f} \tag{10}$$

$$T_2 = \frac{kCAR^{0.63} S_f^{0.54} \Delta t}{W} \tag{11}$$

Both equations contain a mixture of constant and variable parameters. In both equations the constant parameters can be grouped together and expressed as the multiplying coefficient to reduce the computational complexity. Two multiplying coefficients, M_1 and M_2, are calculated for each pipe. Different coefficients are used for each pipe to incorporate the different characteristics such as the roughness and diameter. One constant is used for the Manning's equation, M_1, while the other is used in the Hazen–Williams equation, M_2. These constants stay the same throughout the calculation and are determined using Equations (12) and (13)

$$M_1 = \frac{\Delta t}{Wn} \tag{12}$$

$$M_2 = \frac{\Delta t k A C}{W} \left(\frac{D}{4}\right)^{0.63} \tag{13}$$

where D = pipe diameter (m).

The coefficient M_2 is used when the Hazen–Williams equation is applicable, that is, when full pipe flow occurs resulting in the wetted perimeter covering the whole circumference of the pipe. Thus the hydraulic radius is replaced by $D/4$ in Equation (13). This term could easily be adapted to calculate the hydraulic radius for alternatively shaped pipes if it is required. The multiplying coefficient gives the Transition Rules (14) and (15). It is from these rules the flow rate is determined at each time step

$$T_1 = AM_1R^{2/3}\sqrt{S_f} \tag{14}$$

$$T_2 = M_2S_f^{0.54} \tag{15}$$

Both Transition Rules involve a fewer number of parameters than the original rules, thus lower computational cost is incurred from solving them. This reduces the computation time and improves the model's efficiency. The first Transition Rule, Equation (14), is the more complex of the two as only the block size, length of time step and the Manning's roughness coefficient are constant. The other parameters depend on the water depths at both ends of the pipes. The cross sectional area and hydraulic radius of the flow for each upstream section also need to be calculated, which is time consuming as power functions are involved in the calculations. The second Transition Rule, Equation (15), is much simpler as only the hydraulic gradient varies.

Look-up tables

The multiplying coefficients, Equations (12) and (13), are introduced to reduce the complexity of the model and further improve the modelling efficiency by using the look-up tables, which are calculated only once at the beginning of the simulation run. This reduces the number of calculations required during the simulation. These tables contain the information about the number of blocks that can move for various combinations of depths. The range of depths is taken from the invert and the terrain levels, both of which are known, of neighbourhood manholes. The range is split into equally sized discrete intervals. The number of discrete intervals depends on the degree of accuracy required, the higher accuracy desired the more discrete intervals that are needed. The size of discrete interval used for each manhole is calculated using Equation (16). This prevents very few discrete intervals being used for a small manhole, reducing the accuracy of the associated pipes, or many discrete intervals being used for a large manhole, which would increase the memory usage and increase the computation time

$$\text{Size of discrete interval} = \frac{\text{Total depth of manhole}}{\text{No. discrete intervals desired}} \tag{16}$$

A vector is created containing elements ranging from the invert to the terrain level at a rate of the discrete interval size. For example, for a 2 m deep manhole with water level ranging from the invert at 49 m.a.o.d. (metres above ordinance datum) to the ground at 51 m.a.o.d. and a desired number of discrete intervals of 100 gives a discrete interval size of 0.02 m. The levels, in one manhole, then used to calculate the gradient are

$$[\ 49.00,\quad 49.02,\quad 49.04,\quad \ldots\quad 50.96,\quad 50.98,\quad 51.00\ ]$$

The gradients for all possible combinations of discrete depths are then calculated. The gradient of the pipe is not simply used at all times to allow surcharging and downstream conditions to be taken into consideration. A number of negative gradients are also calculated and these are replaced by the gradient of the pipe. This is used currently as the model cannot simulate reverse flow. The model should not select the solutions of the transition rules produced using negative gradients. However, when solving the rules using negative hydraulic gradients complex numbers may occur within the look-up tables or the program may fail due to unsolvable calculations occurring. To avoid this negative gradients are replaced by 0 resulting in the solution of the transition rules being 0. These gradients are then stored within a matrix. For the free surface flow, only gradients obtained when the upstream and downstream depths are below 95% of the pipe diameter (i.e. not surcharged) are used for the creation of the look-up table for the first Transition Rule (i.e. the Manning's equation). This reduces the memory use of the model and improves the

efficiency. The look-up tables are then referred to during the simulation to obtain the maximum number of blocks allowed to move. To further reduce the memory use of the model, only those gradients calculated when upstream depth has increased and the downstream is at the invert are used to create the look-up table for the second transition rule.

During the simulation, the look-up table for a pipe is referred and the number of movable blocks is obtained. When the flow is not surcharged the depth within each manhole is changed to an integer value. The integer values are obtained by dividing the depth by the discrete interval size. This number is then rounded to the nearest integer. Both the upstream and downstream depths are changed to integer values and together they give the reference for the relevant look-up table element to obtain the number of blocks which can move. If the flow is surcharged the difference in water level between both manholes is found and this is then converted to an integer by the same process as for non-surcharged flow. This value alone is then used as the reference in the look-up table associated with the pipe. If the difference is greater than the depth of the manhole, the latter is simply used. This is done due to the limited size of the look-up tables. Without doing this, the depth would be limited to the top of the manhole which would result in the wrong flow rate being obtained from the look-up tables.

The simulation

When the program is run, the steps shown in Figure 4 are carried out. These can be split into two main steps. The first is the initialisation of the model where the network data is read into the program and using this information the look-up tables are calculated. The next main step is the actual time loop in which the simulation takes place. Both steps will now be looked at in more detail.

Step 1: Initialisation of the model

The procedure (Figure 4) starts when the inflow and network data are read into the program and stored. Each parameter and characteristic is stored in separate vectors to allow for easy accessing during the simulation. The look-up tables are created according to the input data and the methodology described in the previous section. When the depth is small, low velocities occur which cause long travel times between manholes. The travel times can become so high that they are impracticable so a maximum travel time is set. It is set as the time required for a block to travel the length of the pipe at the depth equal to 10% of the pipe diameter. The depth of 10% was chosen as it was found through numerical experiments to be small enough to allow the flow to move slowly through the pipe but still reach the neighbour before the end of the simulation. If this is not included, small amounts of slow moving flow would be left in the pipes at the end of the simulation affecting the mass conservation.

Step 2: The time loop

The time loop starts after the creation of the look-up tables. At the beginning of each time step, blocks arriving from the upstream pipe and/or the surface are added to any blocks already at the manhole. The depth within the manhole is then calculated and taken as the state of the cell. Then, the number of movable blocks is determined from the look-up tables. This is possible as the discharge within a pipe depends on the hydraulic gradient, which is calculated from the water levels at the upstream and downstream manholes.

When the downstream node is surcharged the downstream level is taken to be the downstream water head. However, if the downstream depth is below the pipe soffit, the downstream level is calculated depending on whether the flow is subcritical or supercritical. The flow condition is determined from the depth, y, and critical depth, y_c, in the manhole. The critical depth is calculated using Equation (17), which is a version of Straub's Equation (Straub 1978) for the critical depth. A simplified approach to determining the condition of the flow is taken to help ensure a low computation cost allowing quick simulation times

$$\frac{y_c}{D} = 0.567 \frac{Q^{0.506}}{D^{1.264}} \tag{17}$$

If the depth is greater than the critical depth then the flow is subcritical and the downstream conditions affect the flow rate. While the flow is subcritical, the downstream

level is given by Equation (18). This equation (Akan & Houghtalen 2003) was chosen to ensure the stability of the model. Like all other approximate equations for the critical depth, it has a limited applicability. It is only applicable when inequality, Equation (19), is satisfied. If this inequality is not satisfied, the water level and pipe invert are used as the water level

$$h_{DS} = I_D + \max\left\{y, \frac{y_c + D}{2}\right\} \tag{18}$$

$$0.02 < \frac{y_c}{D} < 0.85 \tag{19}$$

where I_D = downstream invert (m); h_{DS} = water level at the DS(m).

When the critical depth is greater than the depth in the downstream node, the flow is supercritical. When this is the case the downstream conditions can be neglected.

The upstream water level is calculated using Equation (20). This provides a good estimate of the hydraulic grade line and helps to ensure the hydraulic gradient remains positive. The headloss is then revaluated, Equation (21), using the new water levels at the upstream and downstream. The new headloss is then used in the next time step

$$h_{US} = \max\{y + I_U, h_f + y_c + I_D\} \tag{20}$$

$$h_f = \Delta h = h_{US} - h_{DS} \tag{21}$$

where h_f = headloss (m); h_{US} = upstream water level (m); I_U = invert at upstream (m).

It is from these water levels that the matrix element which gives the current number of blocks that can move is determined, as discussed earlier. This number of blocks is placed within a temporary matrix which allows transporting the blocks to the receiving manhole at the appropriate time step. Which look-up table is used depends upon the depth at the upstream end of the pipe. The tables based on Transition Rule (15) are used when the flow is surcharged, whereas the tables created using the Transition Rule (14) are utilised if the flow is not surcharged. The pipe is considered to be surcharged if the water depth at the upstream manhole is above 95% of the diameter of the pipe.

The number of blocks allowed to move is then changed into a velocity. This is done by multiplying the number of blocks by the block size to obtain a volume and then dividing this by the length of the time step in use and the cross area of the flow. The number of blocks that the pipe has space for is then calculated, which restricts the maximum number of blocks to move through the pipe. This is done to ensure the flow volume within the pipe is not greater than the actual capacity of the pipe. After the blocks have been removed from each manhole, the model returns to the beginning of the time loop. Each manhole is simulated independently of every other manhole and does not use information from previous time steps, which is an important feature of CA.

Travel time calculations

The travel time between two adjoining cells is obtained by using Equation (22), which gives a dimensionless travel time when using SI units. This incorporates the time the flow takes move through the pipes into the downstream cell and prevents the flow moving immediately between cells

$$\text{Travel time} = \frac{(L/V)}{\Delta t} \tag{22}$$

where V = velocity of flow (m s^{-1}); L = length of pipe (m).

For simplicity and computational efficiency, an integer travel time is used here. This can, however, result in blocks being included prior to when they should actually arrive at the node. To prevent this, both the real and integer travel times are calculated. The blocks are included in the depth calculation of downstream node only if they arrived at the node prior to the time step in terms of their real travel time. This process results in a more realistic depth at a node to be calculated thus allowing a more accurate flow rate to be determined from the look-up tables. The model only permits the volume included in the depth to be available to move to the downstream node. The remaining blocks are stored in the manhole they are currently in to be included again at this manhole at the subsequent time step.

The travel time is reassessed each time step after the new hydraulic gradient is calculated. As the hydraulic

gradient increases, the number of blocks allowed to move increases and the travel time decreases. From Equation (22) for a very high velocity, the travel time would get very short. This means that it can become less than one which could result in blocks being in multiple cells in a single time step. To prevent this, a minimum travel time of one time step is used.

CASE STUDY

BCA1D was initially tested on a small theoretical network (Austin *et al.* 2013) which showed the model was able to simulate branches merging in and out of a main trunk as well as open channel flow due to low flow rates occurring in the upstream pipes of the network. This network, however, is very small so the model must be tested on a larger and more complex network to determine if the model is a viable alternative to the SVE for sewer flow modelling. A case study was carried out using the sewer network in the Stockbridge area of Keighley, Yorkshire (UK), which is shown in Figure 5. The catchment covers approximately 0.24 km^2 and the system consists of 89 pipes and 90 nodes (manholes). This was slightly reduced to 80 pipes and 81 nodes to improve the overall stability of all models due to a number of undersized pipes that had much smaller diameters in relation to those connected to them. Much of this network is extremely flat as a number of the pipes have a gradient of zero. Examples of the range of pipe characteristics that occur in the network are shown in Table 1. This case study has been selected due to the high number of low gradient pipes making the inclusion of a downstream condition important to ensure a high level of accuracy. It was also selected as its size and complexity are such that it represents a serious test of BCA1D modelling capabilities, but it is still small enough to be able to determine the cause of most problems within the network as the results could be traced upstream.

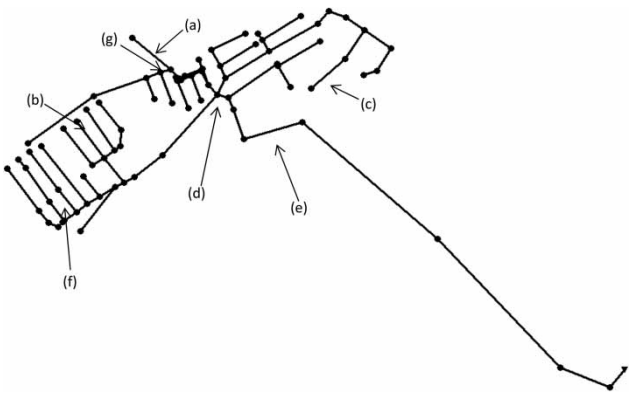

Figure 5 | Schematic of the Keighley sewer network.

Table 1 | Pipe characteristics

Pipe	Pipe ID	Length (m)	Gradient	Diameter (m)
(a)	1	80.85	0.0019	1.125
(b)	27	79	0.0007	0.305
(c)	80	78	0.0054	0.225
(d)	4	18	0.0010	1.145
(e)	7	98	0.0011	1.500
(f)	62	56	0.0000	0.915

The network is simulated using BCA1D, SIPSON (Djordjevic *et al.* 2005) and SWMM5 (Rossman 2010). Both SWMM5 and SIPSON are hydrodynamic models that use the full SVE. This allows BCA1D to be compared to two recognised benchmarks and its level of accuracy can be determined. The storm event used was 3 hours in duration with time-varying intensity and had a total depth of 0.016 m along with a small base flow entering at the most upstream manhole of each branch. The runoff simulation was done by BEMUS (Djordjević *et al.* 1999) and applied to all four models.

RESULTS

The simulation was carried out on a standard standalone 32 bit, 2.66 GHz processor PC. The complete Keighley network was simulated over a period of 5 hours. The simulation time for each model is shown in Table 2. From Table 2, it can be

Table 2 | The computation time

Model	Time step size (s)	Language	Computation time (s)
SWMM 5	30	C++	1.95
SIPSON	5	Fortran	20
BCA1D	30	Matlab	0.87
BCA1D	30	C++	0.37

seen that both versions of BCA1D are faster than both SWMM 5 and SIPSON. This is achieved even in the version created in Matlab. The Matlab version of BCA1D has a higher computation time than the C++ versions as it has to be interpreted each time before it is run which is not the case for a compiled program. This shows the large difference in speed which can occur between compiled and translated. The Matlab version of BCA1D manages to be faster than the hydrodynamic models due to its simplicity as very few computations are required during the simulation. The C++ version of BCA1D is approximately twice as fast as the Matlab version and over five times faster than the SWMM 5 version.

A selection of the pipe simulation results are shown in Figures 6(a)–6(f). The details of these pipes are shown in Table 1 and are also indicated in Figure 5. Those pipes were selected as representative of the complete network. Figures 6(a)–6(c) show a selection of pipes with very low flow.

In all pipes, both versions of BCA1D produces good results on visual comparison with the hydrodynamic models. In pipe (a) the greatest differences occur between the models. During the simulation produced by SIPSON, there is a large negative discharge in the pipe and a very small one in SWMM5 due to a much higher downstream water depth. The high depth in the downstream manhole is occurring due to flow arriving from the larger branch (pipe (g)) also connected to this manhole. This branch is comprised of pipes with much smaller diameters than that of pipe (a) and the final pipe within it becomes surcharged causing the flow to be pressurised and move much quicker than that in pipe (a). In pipe (c) the flow rate is again low. SWMM5 and SIPSON results exhibit sharp dips in the flow rate near the start of the simulation and spikes again later in the simulation due to reverse flow occurring in the pipe. BCA1D also shows small signs of instabilities at these points in the simulation. All four models struggle to simulate this pipe and show signs of instabilities. In pipes (a) and (b), both CA models show small level oscillations being caused by the velocity fluctuating due to changes in depth and flow cross area each time step. The flow results shown in Figures 6(d)–6(f) are very good on visual comparison with all models producing similar results to one another. However, in these pipes there are very small oscillations in the results produced by both versions of BCA1D. This happens because the flow is surcharging and running at full flow. However, due to the limited capacity of the pipes the flow rate has to be reduced briefly. This dip in the flow rate in turn causes a dip in the volume within the pipe allowing the flow rate to rise again the following time step. This pattern causing the oscillations continues until the pipe is no longer surcharged.

To determine the degree of accuracy of BCA1D, the RMSE has been calculated according to Equation (23). The RMSE is an Error Index and a special case of the average error (Willmott *et al.* 1985). The closer the RMSE is to 0, the more similar are the two sets of results being compared. This Index gives a good estimate of the error of the result, however, it does not show if the results are over or underestimated. The NRMSE has been calculated using Equation (24). A NRMSE of 0 indicated perfect results while a value of 1 indicates no agreement. The NSE was also calculated using Equation (25). The NSE lies between $-\infty$ and 1 with the optimal value being 1. If the NSE is negative it is considered poor, whereas if it is positive it can be considered to be good (Moriasi *et al.* 2007)

$$\mathrm{RMSE} = \sqrt{\left[\frac{\sum_{i=1}^{N}(O_{1i} - O_{2i})^2}{N}\right]} \tag{23}$$

$$\mathrm{NRMSE} = \frac{\mathrm{RMSE}}{\max(O_{2i}) - \min(O_{2i})} \tag{24}$$

$$\mathrm{NSE} = 1 - \left[\frac{\sum_{i=1}^{n}(O_{1i} - O_{2i})^2}{\sum_{i=1}^{n}(O_{1i} - \overline{O_1})^2}\right] \tag{25}$$

where N = number of elements in data set (–); O_1 = Model 1 output (–); O_2 = Model 2 output (–).

The RMSE values for the results produced by BCA1D between SIPSON and SWMM 5 have been calculated for each pipe within the network. Similarly, the RMSE between SIPSON and SWMM 5 was also calculated. The RMSE for each of the pipes is shown in Figure 7.

Table 3 contains the maximum RMSE between the models out of all the pipes over the complete network. From this table, it is clear there is little difference between

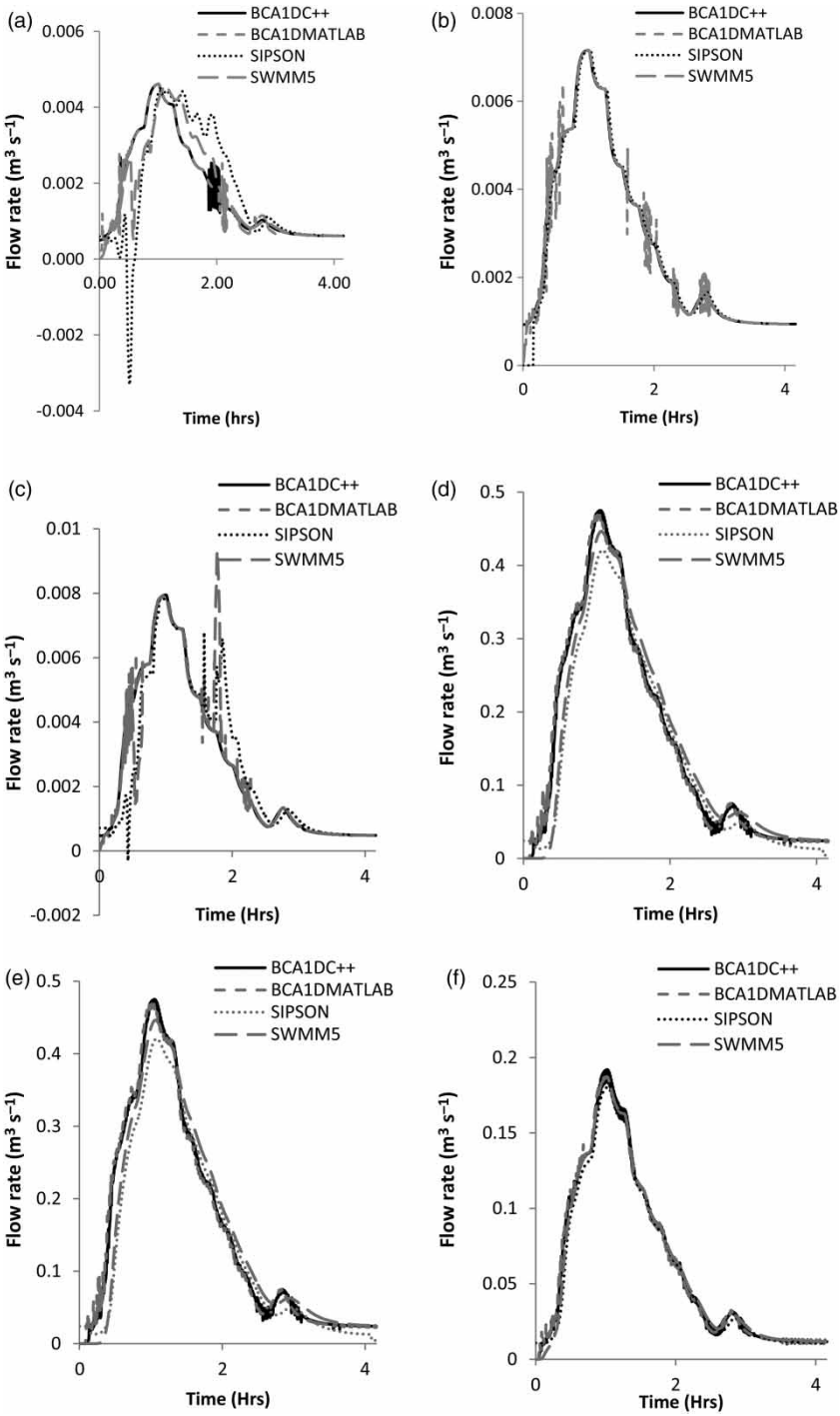

Figure 6 | Flow rate in pipes (a)–(f).

the two versions of BCA1D and both SIPSON and SWMM5, with the maximum being below 0.2. The minimum RMSE for all model comparisons are also very low with the highest being only 2.98e-4. These RMSE results show that there is a good agreement between all the models across the network.

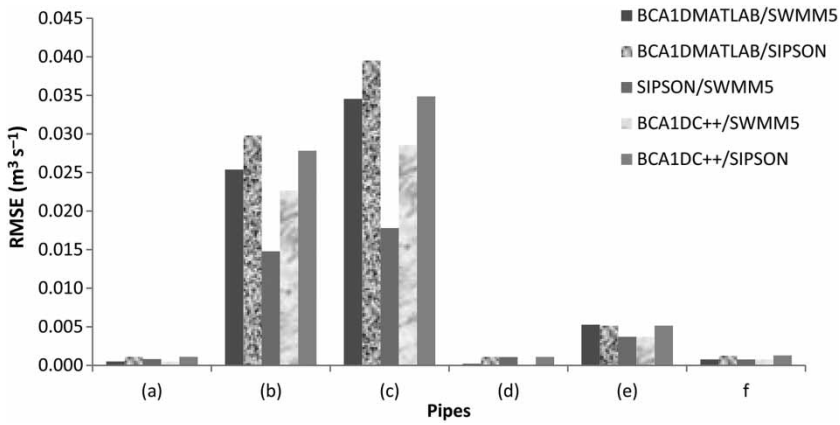

Figure 7 | RMSE in a selection of pipes.

Table 3 | Maximum values of the RMSE

Simulated	Maximum RMSE
BCA1D Matlab/SWMM5	0.081
BCA1D Matlab/SIPSON	0.068
SIPSON/SWMM5	0.032
BCA1D C++/SWMM5	0.074
BCA1D C++/SIPSON	0.065

The next measure looked at is the NRMSE found for the pipes whose flow rates are plotted in Figure 8. From this figure it is clear that low NRMSE were found throughout the network. The maximum NRMSE are shown in Table 4. The maximum is also low, always being below 0.4. An important observation is also that the highest occurs between SIPSON and SWMM5 at a value of 0.341. This emphasises that the hydrodynamic benchmarks vary and do not produce identical results. A similar NRMSE also occurs between both versions of BCA1D and each benchmark with it approximately being 0.17 between BCA1D and SWMM5 and 0.27 between BCA1D and SIPSON.

The maximum and minimum NSE between all the models are shown in Table 5 and a selection of NSE are shown in Figure 9. The maximum NSE between all models is above 0.99 and thus shows a good level of agreement between the models. The minimum between SWMM5 and both other models, however, is negative or zero, showing that at times there is very little agreement.

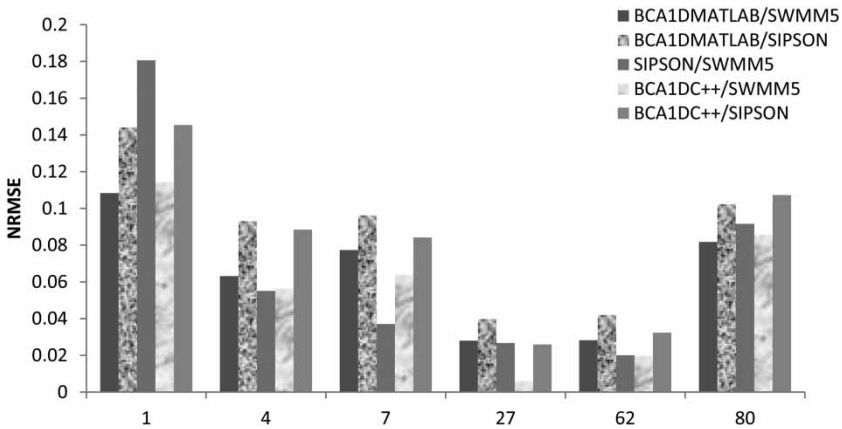

Figure 8 | NRMSE in a selection of pipes.

Table 4 | Maximum values of the NRMSE

Simulated	Maximum NRMSE
BCA1D Matlab/SWMM5	0.177
BCA1D Matlab/SIPSON	0.264
SIPSON/SWMM5	0.341
BCA1D C++/SWMM5	0.170
BCA1D C++/SIPSON	0.267

Table 5 | Minimum and maximum values of the average NSE

Simulated	Minimum NSE	Maximum NSE
BCA1D Matlab/SWMM5	0.5029	0.99968
BCA1D Matlab/SIPSON	–0.2163	0.99898
SIPSON/SWMM5	–0.2167	0.99875
BCA1D C++/SWMM5	0.6660	0.99991
BCA1D C++/SIPSON	–0.2152	0.99925

The maximum RMSE is below 0.01, suggesting a high level of agreement between and BCA1D and the full hydrodynamic models. The NRMSE between both versions of BCA1D and the benchmarks is below 0.1, again suggesting a high level of agreement between the models. The lowest NSE is around –0.2 between both versions of BCA1D and SIPSON, which is similar to the minimum between SIPSON and SWMM5. The spatial and temporal average RMSE, NRMSE, and NSE for each combination of models is presented in Table 6. These were calculated to help determine the overall level of agreement between the models.

The average RMSE in all comparisons is below 0.03, showing agreement between all the models. When comparing the NSEs, like with the RMSEs, BCA1D Matlab and SWMM5 are the closest with BCA1D C++ and SWMM5 only having a slightly lower NSE. The NSE between BCA1D and SIPSON are all above 0.8 showing agreement with SIPSON. The average NRMSE is also low as it is below 0.08 indicating a high level of agreement between all the models. The NRMSE that occurs between both versions of BCA1D and SIPSON is significantly higher than that between both versions of BCA1D SWMM5. The NRMSE between BCA1D and SWMM5 shows a similar agreement between SIPSON and SWMM5. The variation in level of agreement between the models is caused by both metrics being more sensitive to different types of errors. This emphasises the importance of using more than one metric to analyse the results. The average RMSE for both versions of BCA1D is almost identical and the NSE is acceptable for when they are compared to either SWMM 5 or SIPSON. This shows that there is a similar level of agreement between both versions of BCA1D and the hydrodynamic models and they can all be considered to be in good agreement with one another.

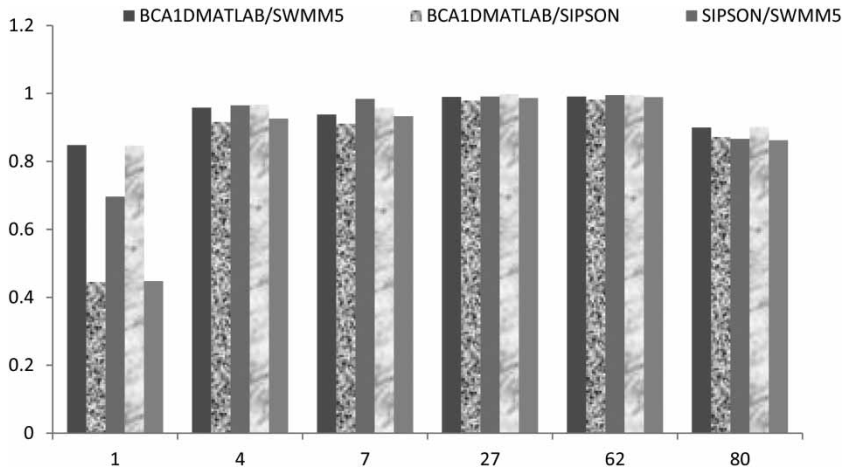

Figure 9 | NSE in a selection of pipes.

Table 6 | Spatial and temporal average of the RMSE, NRMSE and NSE

Simulated	Predicted	Ave. RMSE	Ave. NRMSE	Ave. NSE
BCA1DMATLAB	SWMM 5	0.006	0.050	0.946
BCA1D MATLAB	SIPSON	0.009	0.079	0.876
SWMM 5	SIPSON	0.005	0.053	0.928
BCA1D C++	SWMM 5	0.006	0.041	0.959
BCA1D C++	SIPSON	0.008	0.0722	0.887

CONCLUSIONS

In this study, the methodology for a new CA based sewer model, BCA1D, has been presented. This model represents each manhole as a cell and moves the flow in the form of blocks. It uses either the Manning's equation or the Hazen–Williams equation, to determine the number of blocks which can move in each time step. To improve the efficiency of the model, the flow rate is obtained from the look-up tables developed using either of the flow equations. To obtain the value from the look-up tables, the upstream and downstream water levels are calculated at each time step.

This new model, along with SWMM5 and SIPSON, was used to simulate the sewer network in Keighley, Yorkshire. From the visual comparison, all models produce similar results except when reverse flow becomes important. This is a weakness in the new model as it does not currently take these conditions into account. Including the ability to simulate these effects is the next step in the development of the BAC1D model. The model, however, is sufficiently accurate as it is shown by the low RMSE values (close to zero) and high NSE values (close to one) in comparison to both fully hydrodynamic models, SWMM5 and SIPSON, thus it can be considered to be in good agreement with both. BCA1D was also found to be faster than both SWMM5 and SIPSON. The saving in simulation time is more significant in tests on larger networks (not presented here). Future work will also include expanding the model to simulate a wider variety of structures found in sewer networks.

ACKNOWLEDGEMENTS

The authors would like to acknowledge the funding provided by the UK Engineering and Physical Sciences Research Council, grant no. GR/J09796 (Cellular Automata Dual Drainage Simulation, CADDIES).

REFERENCES

Achleitner, S., Moederl, M. & Rauch, W. 2007 CITY DRAIN (c) – An open source approach for simulation of integrated urban drainage systems. *Environ. Model. Softw.* **22**, 1184–1195.

Akan, O. & Houghtalen, R. 2003 *Urban Hydrology, Hydraulics and Stormwater Quality: Engineering Applications and Computer Modeling*. John Wiley & Sons, Chichester.

Albert, J. & Culik, K. 1987 A simple universal cellular automaton and its one-way and totalistic version. *Complex Syst.* **1**, 1–16.

Austin, R., Chen, A., Savic, D. & Djordjevic, S. 2013 *Fast Simulation of Sewer Flow using Cellular Automata*. Presented at the Novatech, Lyon, France.

Bates, P. D., Horritt, M. S. & Fewtrell, T. J. 2010 A simple inertial formulation of the shallow water equations for efficient two-dimensional flood inundation modelling. *J. Hydrol.* **387**, 33–45.

Calabrò, P. S. 2001 Cosmoss: conceptual simplified model for sewer system simulation: a new model for urban runoff quality. *Urban Water* **3**, 33–42.

Chatila, J. G. 2003 Muskingum Method, EXTRAN and ONE-D for routing unsteady flows in open channels. *Can. Water Resour. J.* **28**, 481–498.

Djordjević, S., Prodanović, D. & Maksimović, Č. 1999 An approach to simulation of dual drainage. *Water Sci. Technol.* **39**, 95–103.

Djordjevic, S., Prodanovic, D., Maksimovic, C., Ivetic, M. & Savic, D. 2005 SIPSON – Simulation of interaction between pipe flow and surface overland flow in networks. *Water Sci. Technol.* **52**, 275–283.

Dottori, F. & Todini, E. 2010 A 2D Flood Inundation Model Based on Cellular Automata Approach. In *XVIII International Conference on Water Resources*, Barcelona.

Dottori, F. & Todini, E. 2011 Developments of a flood inundation model based on the cellular automata approach: testing different methods to improve model performance. *Phys. Chem. Earth A/B/C* **36**, 266–280.

Durand, B., Formenti, E. & Róka, Z. 2003 Number-conserving cellular automata I: decidability. *Theor. Comp. Sci.* **299**, 523–535.

Evans, E. & Office of Science and Technology 2004 *Foresight: Scientific Summary. Volume I, Future Risks and their Drivers*. Office of Science and Technology, London.

Fuk, H. 2004 Probabilistic cellular automata with conserved quantities. *Nonlinearity* **17**, 159–173.

Fullerton, J. 2004 *A Simplified Modelling Approach for Storm Water Flow Optimization*. University of Exeter, Exeter.

Gardner, M. 1970 Mathematical games: the fantastic combinations of John Conway's new solitaire game 'life'. *Sci. Am.* **223**, 120–123.

Ghimire, B., Chen, A. S., Guidolin, M., Keedwell, E. C., Djordjević, S. & Savić, D. A. 2013 Formulation of a fast 2D urban pluvial flood model using a cellular automata approach. *J. Hydroinform.* **15** (3), 676–686.

ITWH 1995 *KOSIM*. Institut für Technisch-Wissenschaftliche Hydrologie, Hannover.

McCarthy, G. 1938 The Unit Hydrograph and Flood Routing. *Proceedings of the Conference of North Atlantic Division*, US Army Corps of Engineers. New London, Cronin.

Meirlaen, J. & Vanrolleghem, P. 2000 Simulation of the Integrated Urban Wastewater System Using Mechanistic Surrogate Models. In *Monitoring and Modelling Catchment Water Quantity and Quality (8th General Assembly of the European Network of Experimental and Representative Basins, ERB2000)*. Gent, Belgium.

Meirlaen, J., Schutze, M., Van der Stede, D., Butler, D. & Vanrolleghem, P. 2000 Fast, parallel simulation of the integrated urban wastewater system. In: *Proceedings WAPUG Spring meeting*. Birmingham, UK, May 9, 2000, p. 9.

Miyamoto, H. & Sasaki, S. 1997 Simulating lava flows by an improved cellular automata method. *Comp. Geosci.* **23**, 283–292.

Moriasi, D. N., Arnold, J. G., Van Liew, M. W., Bingner, R. L., Harmel, R. D. & Veith, T. L. 2007 Model evaluation guidelines for systematic quantification of accuracy in watershed simulations. *Trans. ASABE* **50** (3), 885–900.

Morita, K. 2012 Universality of One-Dimensional Reversible and Number-Conserving Cellular Automata. arXiv preprint arXiv: 1208.2760.

Nash, J. E. & Sutcliffe, J. V. 1970 River flow forecasting through conceptual models part I – A discussion of principles. *J. Hydrol.* **10**, 282–290.

Noto, L. & Tucciarelli, T. 2001 DORA algorithm for network flow models with improved stability and convergence properties. *J. Hydraul. Eng. ASCE* **127**, 380–391.

Paulsen, O. 1986 *Kontinuierliche Simulation von Abflüssen und Schmutzfrachten in der Trennentwässerung*. Universität Hannover, Hannover.

Rossman, L. 2010 *SWMM5*. United Sates Environmental Protection Agency, Cincinnati, OH.

Samani, H. M. V. & Jebelifard, S. 2003 Design of circular urban storm sewer systems using multilinear Muskingum flow routing method. *J. Hydraul. Eng. ASCE* **129**, 832–838.

Sanchez, A., Medina, N., Vojinovic, Z. & Price, R. 2014 An integrated cellular automata evolutionary-based approach for evaluating future scenarios and the expansion of urban drainage networks. *J. Hydroinform.* **16** (2), 319–340.

Singh, V. P. & McCann, R. C. 1980 Some notes on Muskingum method of flood routing. *J. Hydrol.* **48**, 343–361.

Solvi, A.-M. 2007 *Modelling the Sewer-Treatment-Urban River System in View of the EU Water Framework Directive*. BIOMATH, Department of Applied Mathematics, Biometrics and Process Control PhD Thesis, Ghent University, Belgium.

Solvi, A.-M., Schosseler, P. & Vanrolleghem, P. A. 2008 Combined Immission-Emission based Evaluation of Integrated Urban Wastewater System Scenarios. *Presented at the 11th International Conference on Urban Drainage*, Edinburgh, Scotland, UK.

Straub, W. O. 1978 A quick and easy way to calculate critical and conjugate depths in circular open channels. *Civil Eng.* **48** (12), 70–71.

Thomas, N. 2000 *Optimal Pollution Control Models for Interceptor Sewer Systems*. University of Liverpool, Liverpool, UK.

Vaes, G. 1997 REMULI: Reservoir Model with Multi-Linear Throughflow Relationships. Report, Hydraulics Laboratory, Kath. Universiteit Leuven, Leuven, Belgium.

Vanrolleghem, P. A., Kamradt, B., Solvi, A. & Muschalla, D. 2009 Making the Best of two Hydrological Flow Routing Models: Nonlinear Outflow – Volume Relationships and Backwater Effects Model. In *8th International Conference on Urban Drainage Modelling*, Tokyo, Japan.

Von Neumann, J. 1966 *Theory of Self Reproducing Automata*. University of Illinois Press, Urbana, IL.

White, I. 2008 The absorbent city: urban form and flood risk management. *Proc. ICE Urban Design Plann.* **161**, 151–161.

Willmott, C. J., Ackleson, S. G., Davis, R. E., Feddema, J. J., Klink, K. M., Legates, D. R., O'Donnell, J. & Rowe, C. M. 1985 Statistics for the evaluation and comparison of models. *J. Geophys. Res.* **90**, 8995–9005.

Wolfram, S. 1982 Cellular Automata as Simple Self-Organizing Systems. Caltech preprint CALT-68-938 5.

Wolfram, S. 1983 Statistical mechanics of cellular automata. *Rev. Mod. Phys.* **55**, 601.

Wolfram, S. 1984 Universality and complexity in cellular automata. *Physica D* **10**, 1–35.

Yang, L., Fang, W., Huang, R. & Deng, Z. 2002 Occupant evacuation model based on cellular automata in fire. *Chin. Sci. Bull.* **47**, 1484–1488.

First received 3 June 2013; accepted in revised form 24 May 2014. Available online 25 June 2014