Exercise 1: Introduction to 2D numerical flood modelling

This exercise is part of a series designed to teach students how floodplain inundation can be simulated by numerical models and how flood risk maps can be produced from simulation results within the KULTURisk framework. In this case the flood inundation model is LISFLOOD-FP (hereafter lisflood), however most two-dimensional hydraulic models can be used in much the same way. This first exercise introduces users to lisflood using very simple test cases to explore the difference between results produced using the various solver options within lisflood.

It may be useful to look at the following papers which describe the development of three of the 2D model solvers used in the exercise (Bates et al., 2010) and from which the test cases are drawn (de Almeida and Bates, in press.) :

DE ALMEIDA, G. A. M. & BATES, P. 2013. Applicability of the local inertial approximation of the shallow water equations to flood modelling. *Water Resources Research,* 49(8), 4833 – 4844.

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. *Journal of Hydrology,* 387**,** 33-45.

Introduction to LISFLOOD-FP

In its most simple form the lisflood model takes a raster Digital Elevation Model and water inflow details (location and rate) and simulates inundation dynamics. The model uses hydraulic continuity principles to calculate the water depth in each cell of the raster grid. Water is routed across the floodplains using a simple storage cell algorithm based on the difference in hydraulic head between adjacent cells (Figure 1).

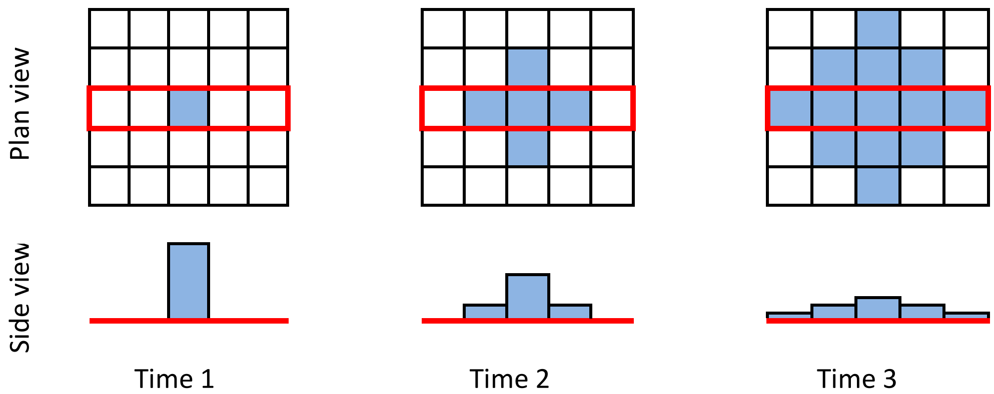

Figure 1. How water flows across the domain by storing water in cells and then allowing water to flow to neighbouring cells in the x-y Cartesian directions over each time-step.

Several solvers are available for calculating how much water will flow between adjacent cells over a given time. In this exercise we will look three different solvers. The first, which will be referred to as the “flow limited” model, uses an approximation of the diffusion wave equations based on the Manning’s equation. It calculates flow between cells during a time-step of fixed duration as a function of the friction slope and water slope. To ensure upstream cells do not completely empty during a time-step (leading to a reversal in flow direction) a limit on the maximum flow volume must be implemented. The second solver (referred to as the "adaptive" model) overcomes the flow reversal problem by using a time-step which varies in time throughout the simulation according to the water slope and grid resolution to ensure cells cannot empty during a time-step. The third solver is the most complex hydraulically and is referred to as the "acceleration" model. It is a simplified form of the shallow water equations, where advection is assumed negligible and calculates flow between cells as a function of the friction slope, water slope and water acceleration. It uses a time-step which varies according to the well known Courant-Friedrichs-Lewy condition. The controlling equations for each of these solvers can be found in the appendix and further explanation is presented in the following papers:

BATES, P. D. & DE ROO, A. P. J. 2000. A simple raster-based model for flood inundation simulation. *Journal of Hydrology,* 236**,** 54-77.

HUNTER, N. M., HORRITT, M. S., BATES, P. D., WILSON, M. D. & WERNER, M. G. F. 2005. An adaptive time-step solution for raster-based storage cell modelling of floodplain inundation. *Advances in Water Resources,* 28**,** 975-991.

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. *Journal of Hydrology,* 387**,** 33-45.

Data provided

All of the input files necessary to run simulations of the following two test cases with a high friction and low friction scenario are provided in the zip file for this exercise. You will also need to have downloaded LISFLOOD from the University of Bristol Hydrology website. In the folders for each test case you will find files providing the data which will be to read in to the simulation: a digital elevation model (DEM), the boundary conditions for the domain (`*.bci`) and details of time-varying boundary conditions (`*.bdy`). Importantly, each folder will also contain a file which instructs lisflood what data to read in and which options to use for the simulation (`*.par`). There will be one `.par` file for each of the model solvers explored in this exercise.

Test Cases

All of the test cases consist of a model domain which is rectangular in plan view, with a time varying water height specified at one boundary. The domain's remaining boundaries are closed (as is default). The following figures illustrate the two test cases.

1. Non-breaking wave propagation over a horizontal plane

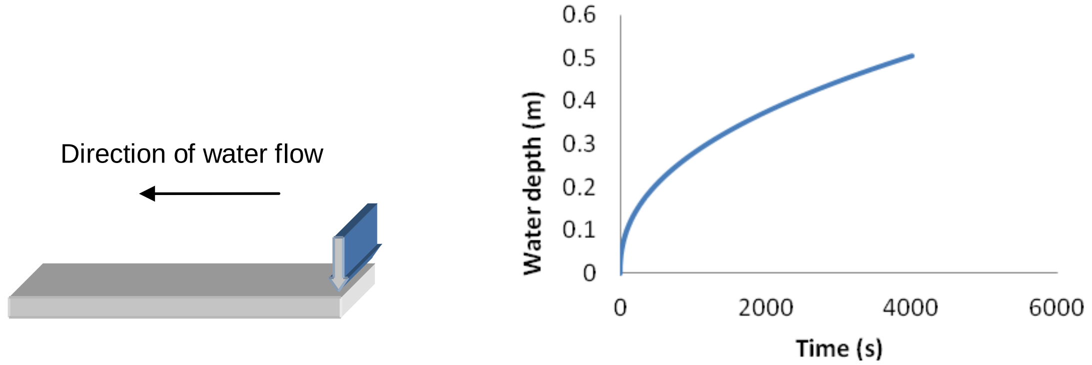

Figure 2 Left panel shows schematic of the model domain for test case 1, location of the time-varying water height boundary indicated by arrow. Right panel shows the time varying water depths specified at the boundary, which is the same for the high and low friction scenario.

2. Non-breaking wave run-up on a planar beach

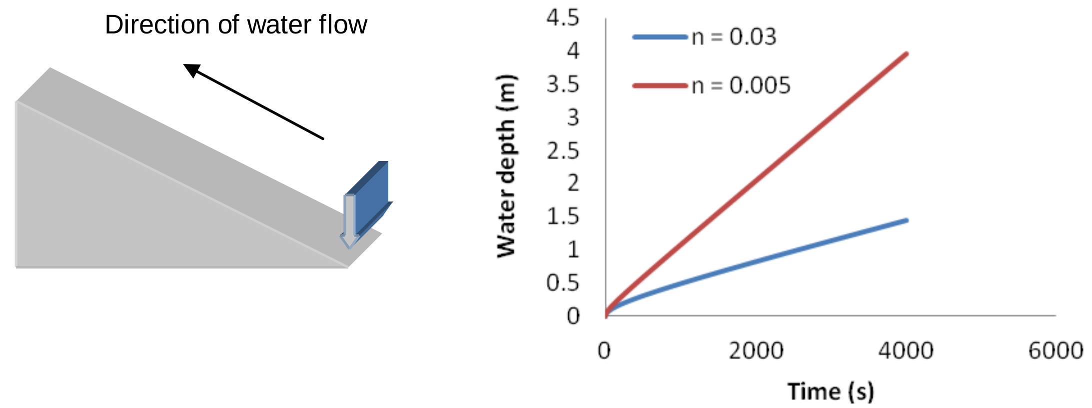

Figure 3 Left panel shows schematic of the model domain for test case 2, location of time-varying water height boundary indicated by arrow. Right panel shows the time varying water depths specified at the boundary for the high and low friction scenario.

Looking in more detail (optional)...

These simulations can be run and the results explored with only a qualitative understanding of these test cases. However for the more quantitative minded, further exploration is possible. The first test case (shown above) can be solved using an analytical solution, allowing calculation of water height (h) at any position along the x axis at any time. The analytical solution is a form of the full 1D shallow water equations (equation 1 in appendix) where a constant water velocity and horizontal bed is imposed. This removes the acceleration and advection terms and simplifies the water slope term such that equation 1 becomes:

$$\frac{\partial h}{\partial x}+\frac{n^2u^2}{h^{\frac{4}{3}}}=0$$

[a]

which can then be re-arranged and integrated to give:

$$h = \left[-\frac{7}{3}(n^2u^2x + C)\right]^{3/7}$$

[b]

If the moving boundary condition of $h_{(ut,t)} = 0$ *is imposed (i.e. water height is zero at the wave front) then C must equal ut and equation [b] becomes:*

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

[c]

The governing equation for test case 2 is similar to equation [a] but for the addition of a bed slope:

$$\frac{\partial h}{\partial x} + \frac{\partial z}{\partial x} + \frac{n^2u^2}{h^{\frac{4}{3}}} = 0$$

[d]

Calculating the analytical solution and boundary conditions for test case 1 (optional):

Equation [c] allows the water depth at any position and time to be calculated, allowing comparison between the analytical solution and the water depth files produced by lisflood to assess the model and calculation of suitable boundary conditions to input to the model. Analytical solutions and boundary conditions for the simulation set-ups provided with this exercise are given in the files analytical.xls and planar.bci respectively, however you could also calculate them yourself using equation [c].

- **Calculate the water depth expected where x = 0 for the high and low friction scenarios using the parameter values given in Table 1. These values will be the boundary conditions used later in the exercise for running the simulations (`.bci` file). Note – the values of n and u have been chosen to give near identical boundary conditions.**

Table 1

Parameter	High friction scenario	Low friction scenario
Time (t)	0 to 3600 seconds in increments of 1 second	0 to 3600 seconds in increments of 1 second
Manning friction coefficient (n)	0.03	0.005
Speed (u, ms^{-1})	0.29	0.95

- **Calculate the water profile expected at t = 3600 seconds for the high and low friction scenarios using the parameter values given in Table 2 . These will be used to assess the model results later in the exercise.**

Table 2

Parameter	High friction scenario	Low friction scenario
Distance (x,meters)	0.5 to 4000.5 in 1 meter increments	0.5 to 4000.5 in 1 meter increments
Manning friction coefficient (n)	0.03	0.005
Speed (u, ms^{-1})	0.29	0.95

Lisflood set-up and use:

1) **Simulation set-up: viewing and understanding the .par files.**

- Take a look at one of the horizontal plane `acceleration.par` files using your favourite text editor.

The model is controlled by the steering files `*.par`. These give the names of key files and controlling parameters. You can see that they are set up for a simulation (`sim_time`) of 3600 s duration and a save interval (`saveint`) of 360 s. In other words, they will produce 10 images of the flood extent developing as the model runs and write results to a text file (`.mass file`) every 10 s (`massint`). These results are saved in the directory (`dirroot`) and with prefix (`resroot`) specified in the .par file. ***Note - the model will overwrite any files with the same names as current model output in the specified folders without giving warning.*** *A spatially uniform Manning's n for the flood plain has been specified (`fpfric`) and the model is instructed to read in the files which specify the location and rate of water input (`bcifile` and `bdyfile` respectively).*

2) **Running a simulation**

This is a research code so it runs in COMMAND PROMPT or UNIX to make Monte Carlo analysis easier.

- Copy and paste the lisflood.exe and .dll files into the folder containing the .par file for the model you want to run
- Open a DOS prompt and move to the directory in which the .par file is located[1].
- To run the model, at the command prompt type:
  `lisflood –v your_chosen_parfile.par`

"-v" stands for verbose mode and is used to increase the amount of information communicated by the code about what it is doing. The code should take a few seconds to run and will have saved the results as specified in the .par file.

3) **Viewing results**

As default the model will output water depth files at each saveint (-xxx.wd), a file containing a variety of diagnostic parameters saved at each massint (*.mass) and files relating to maximum inundation depths (*.max), elevations (*.mxe) and times (*.inittm, *.maxtm and *.totaltm). Additional keywords added to the .par file turn off these outputs or turn on others (see instruction manual for details).*

- Take a look at the .wd and .mass text files in the results directory using a text editor or excel[2]

Water depth files consist of a header containing details of the size and location of the grid, as well as the "NoData" value, followed by a grid of water depth values for each cell in the domain (in this case

[1] To do this open a Command Prompt window: Start > All Programs > Accessories. To move to the right directory type cd (change directory) followed by the full directory name (e.g. cd c:\Documents and SETTINGS\a User\My Documents\Lisflood). If you are not in the right drive you will first need to enter the drive letter followed by a colon.

[2] To open in excel use File > Open and ensure "All files (*.*) is selected as the file type. Then choose "Delimited" and Tab for the delimiter.

the domain is only 1 cell wide). The .mass file consists of a header line followed by 12 columns of data as follows:

Table 3 Details of results saved to `.mass` file

Name	Description
Time	Time in seconds at which data was saved
Tstep	Value of the time-step specified by the user in the par file
MinTstep	Minimum time-step used so far in the simulation
itCount	Number of time-steps since the start of the simulation.
Area	Area inundated in m^2
Vol,e-6	Volume of water in the domain m^3
Qin	Inflow discharge in m^3s^{-1}.
Hds	Water depth at the downstream exit of the model domain
Qout	Calculated outflow discharge at the downstream exit of the model domain in m^3s^{-1}
Qerror	Volume error per second in m^3s^{-1}
Verror	Volume error per mass interval (massint variable in the parameter file) m^3.
Rain-Inf+Evap	Cumulative effect of infiltration, evaporation and rainfall over the simulation in 10^3 m^3.

- Lastly, take a look at the raster files*.max, *.mxe, *.inittm, *.maxtm and *.totaltm. These can be opened in a text editor or in excel.

<u>Exploring LISFLOOD-FP</u>

The data for this practical includes .par files for simulating both test cases under the high and low friction scenarios using several of the 2D model solvers provided within LISFLOOD-FP. First, view and compare the .par files for the different models and scenarios to see how the different solver options are turned on and off using the .par file.

Run each model simulation and take a look at the results (all except the adaptive models should take only a matter of seconds to run[3]). Compare the final water depth results with each other (you could copy them into a single excel file and plot them on the same graph). For the first text case you can also compare results to the analytical solution (provided in the .xls file or calculated yourself). Using this model set-up, which model compares most favourably with the analytical solution? Also, what is the minimum time-step used by each of the solvers and how long does each simulation takes to run. These will be the major considerations facing computer modellers as they decide which method to use. For more details read the 2 papers suggested at the beginning of this document or the introduction section of the Lisflood user manual.

Further options to explore...

Read the user manual to find out in more detail what the files provided are doing and further options for simulation runs. Familiarise yourself with the program: the options available, the format of output and input files and how to manipulate them. Explore the effect of some of these options by either changing the values of model parameters, outputting additional results files, modifying

[3] Due to the shallow water surface gradients and consequently very small time-steps used by the adaptive model, simulations using the adaptive solver may take a LONG time to run. Consider leaving them to run overnight (or to stop the model running type “ctrl-c” at the command line). Alternatively, results files have been provided for these simulations in the answers folder – the low friction simulations took up to 3 weeks to run!

files such as the DEM or boundary condition files, or even creating additional files such as a `manningsfile` to specify spatially varying manning's values.

Here are some suggested ideas (you may have to look up some of these items in the instruction manual). Brief explanations of the expected effects of some of these are given in the answers.xls file.

- Move the location of the point sources to the opposite side of the domain
- Increase `sim_time` – what happens (or doesn't happen) in the horizontal plane test case when the water reaches the opposite side of the domain? Why is this? Can you change this outcome by varying the input files?
- Create a `manningfile` with spatially varying floodplain friction. Try bands of low and high friction values perpendicular to the flow direction
- For the acceleration model look at the effect of setting the value of `theta` to 1 for the low friction scenarios
- Using the flow limited solver, try varying the initial time-step (`initial_tstep`) to try and improve results, or vary the `Qlim` value.
- Vary `fpfric` (other example values could range from ~0.016 for asphalt, 0.035 for short grass and up to 0.15 for wooded floodplain). You could use the equations in the "looking in more detail" exercise box to create boundary conditions and an analytical solution for specific combinations of u and n.

Acknowledgments

The development of this online course was supported by funding through the European Community's Seventh Framework Programme through the KULTURisk Project (ENV.2010.1.3.2-1) and work was carried out by colleagues at the University of Bristol, UK.

Appendix – governing equations for lisflood 2D solvers

In all cases the dynamics of water distribution over the flood plain are simulated by treating each cell in the domain as a storage volume, whose volume change with each time-step is calculated by summing the water fluxes in and out of cell in each of the four Cartesian directions. Here we will briefly describe the main characteristics of the solvers used to calculate water fluxes between cells on the flood plain (2D solvers). For flow in river channels (1D solvers and subgrid) please see exercise 2 and the lisflood instruction manual for details.

Full shallow water equations

All of the solvers used in lisflood are approximations to varying degrees of the full shallow water equations. Continuity and momentum equations for the 1D full shallow water equations are given below (equations **(1)** and **(2)** respectively).

$$\underbrace{\frac{\partial Q}{\partial t}}_{local\ acceleration} + \underbrace{\frac{\partial}{\partial x}\left(\frac{Q^2}{A}\right)}_{convective\ acceleration} + \underbrace{gA\frac{\partial(h+z)}{\partial x}}_{water\ slope} + \underbrace{\frac{gn^2Q^2}{R^{4/3}A}}_{friction\ slope} = 0, \tag{1}$$

$$\frac{\partial A}{\partial x} + \frac{\partial Q}{\partial x} = 0, \tag{2}$$

Governing equations for the flow limited model

The simplest 2D solver is based on the Manning's equation in which flow of water is controlled by only the pressure and bed gradients (the water slope) and the friction. Both local and convective acceleration terms are assumed negligible:

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

This solver uses a user define time-step which is of fixed duration for the whole simulation. However, unless this time-step is very small it may be long enough for all the water to drain from one cell to the next over a single time-step, leading to flow in the opposite direction during the next time-step. To overcome this problem a "flow limiter" was introduced which sets a limit on the volume of water allowed to flow between cells during a single time-step, as a function of flow depth, grid size and time-step:

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

Governing equations for the adaptive model

This solver uses the same equation as the flow limited version to calculate flow between cells on the flood plain (equation (3). However, it differs from the flow limited solver by having a time-step which varies in duration throughout the simulation rather than one with a fixed duration. This overcomes the problem of cells emptying during a time-step without the need of a flow limiter. The equation which governs time-step duration is shown below:

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

(5)

<u>**Governing equations for the acceleration model**</u>

Once again, this solver uses an approximation of the shallow water equations to calculate water flow between cells. In addition to the water and friction slopes, water flow is also a function of local acceleration and only convective acceleration is assumed to be negligible. Water flow is calculated as below:

$$Q_x^{t+\Delta t} = \frac{\overbrace{Q^t}^{local\ acceleration\ term} - \overbrace{gh^t \Delta t \frac{\partial (h_t + z)}{\partial x}}^{water\ slope\ term}}{1 + \underbrace{g \Delta t n^2 \left| Q_{flow}^t \right| / \left[{h_{flow}^t}^{7/3} \right]}_{friction\ term}}$$

(6)

Like the adaptive version, the time-step used with the acceleration solver varies throughout the simulation. In this case it varies according to the cell size and water depth:

$$\Delta t_{max} = \alpha \frac{\Delta x}{\sqrt{gh_t}}$$

(7)