Documentation of *FullSWOF_2D* v1.02.00

Christian Laguerre, Christian.Laguerre@math.cnrs.fr

January 24, 2012

Contents

1 Presentation of the *FullSWOF_2D* software

The name *FullSWOF_2D* stands for “Full Shallow Water equations for Overland Flow in two dimensions of space”. In this software, the Shallow Water (or Saint-Venant) equations are solved using finite volumes and numerical methods especially chosen for hydrodynamic purposes (transitions between wet and dry areas, small water heights, steady preservation ...). For explanations concerning the numerical schemes and approximations, the reader is referred to [3], [4] and [5]. For a precise description of the structure of the software in several classes, see the Doxygen file (refman.pdf in the *doc* directory) and [2]. The structure of the source code is designed to make future evolutions easy: for example, a new friction law can easily be added in the libfriction library, by creating a new friction file.
If you plan to change the code of *FullSWOF_2D*, see Section 5 for explanations on how to use the benchmarks. Doing so, you should pay attention to the license (section 2.2)

2 Software distribution

2.1 How to download *FullSWOF_2D*?

The *FullSWOF_2D* software can be downloaded on the website `https://sourcesup.cru.fr/projects/fullswof-2d/`

2.2 License

This software is distributed under CeCILL-V2 (GPL compatible) free software license. So, you are authorized to use the Software, without any limitation as to its fields of application.

If you make changes to *FullSWOF_2D* code, you are welcome to **contribute your changes to the main repository**, directly through the website (`https://sourcesup.cru.fr/projects/fullswof-2d/`) or by contacting its main developer (Christian Laguerre, Christian.Laguerre@math.cnrs.fr). You may prefer to distribute yourself the *Modified Software*. In such a case, we ask you to **change its name** in order to avoid confusion between your software and the original one. In such a case, pay attention to the text that follows.

The license authorizes you to distribute the *Modified Software*, in source code or object code form, provided that said distribution complies with all the provisions of the *Agreement* and is accompanied by:

- a copy of the Agreement,
- a notice relating to the limitation of both the Licensor’s warranty and liability,

and that, in the event that only the object code of the *Modified Software is redistributed*, you allows future users *access to the full source code of the Modified Software by indicating how to* *access it*, it being understood that the additional cost of acquiring the source code shall not exceed the cost of transferring the data.

For further explanation about this free software license, you should read the following links:

- `http://www.cecill.info/licences/Licence_CeCILL_V2-fr.html`
- `http://www.cecill.info/licences/Licence_CeCILL_V2-en.html`

2.3 Installation

First unzip the archive of the software. When you are in the *FullSWOF_2D* directory, write the following lines:

```
make cleanall
make
```

For the first tests, you can use the *Exp01* directory, where you will find examples of inputs files:

```
cd Exp01
../bin/FullSWOF_2D
```

3 Input and output values

When launched, *FullSWOF_2D* expects two subdirectories: one for the inputs, one for the outputs. In the following sections, the notation `<x>` stands for the tag corresponding to the x variable, whereas the square brackets [$\cdot$] give the unit of the variable.

3.1 *Inputs* directory

You can set the values of most of parameters in the *parameters.txt* file, located in the *Inputs* directory (the values we advise to use for overland flow can be found in Section 3.1.12).

3.1.1 Space and time scales

First, you have to specify the **number of grid cells** `<Nxcell>` and `<Nycell>` (in space) and the **length of the domain** along x and y (`<L>` [m] and `<l>` [m], respectively). For the time, you should set the value of the **duration of the simulation** `<T>` [s].

You can run *FullSWOF_2D* either with a **constant CFL** `<cflfix>` or with **constant time step** `<dtfix>`. The type of time constraint is defined by <scheme_type>. You can choose either:

- case 1: fixed CFL value. If you choose this case, usually, a CFL value equal to 0.4 is suitable.
- case 2: fixed time step (in seconds).

If you choose a constant CFL, you have to specify the value of `<cflfix>` only. If you choose a constant time step, you have to specify the values of both `<dtfix>` and `<cflfix>`). Indeed, it is always necessary to set a CFL value even if you choose the fixed time step because FullSWOF verifies at each loop in time that the time step is not greater than this CFL value in order to respect the stability condition.

3.1.2 Boundary conditions

For some boundary conditions, you may have to specify the discharge and the water height on the boundary. You must use the following principles:

Imposed discharges `<left_imp_discharge>`, `<right_imp_discharge>`, `<bottom_imp_discharge>` and `<top_imp_discharge>` [m^3/s] correspond to the discharges at the left, right, bottom and top boundaries, respectively.
At the left (*i.e.* $x = 0$) and bottom (*i.e.* $y = 0$) boundaries, if you want an incoming flow, you have to impose a positive discharge whereas for an outgoing flow you must impose a *negative* discharge.
At the opposite boundaries (*i.e.* $x = x_{max}$ and $y = y_{max}$), impose a *negative* discharge for the inflow and a positive discharge for the outflow;

Imposed water heights `<left_imp_h>`, <right_imp_h>, `<bottom_imp_h>` and `<top_imp_h>` [m] correspond to the water heights at the left, right, bottom and top boundaries respectively.

You must choose the **left, right, top** and **bottom boundary conditions** (`<Lbound>`, `<Rbound>`, `<Tbound>` and `<Bbound>`):

- the case 1 corresponds to the liquid boundary condition (based on the modified characteristic method). **Imposed discharge and water height** have to be specified. Depending on the regime (sub/super-critical) and on the boundary (inflow/outflow), the value of the imposed discharge may not be used by *FullSWOF_2D*;
- case 2 is for the wall condition;
- the case 3 is for the Neumann boundary condition, which means that the normal derivatives of the water height and velocity are null. Physically, this case represents an open boundary;
- the case 4 stands for the periodic case, in which the outflow on one boundary is considered as the inflow on the other boundary. In this case, at each time step, we impose the incoming flow to be equal to the outgoing flow;
- in case 5, the value of the discharge is imposed. You must specify the **value of flow discharge** and the **specific water height**, but the latter will be considered only in supercritical cases.

3.1.3 Friction law

You also have to impose the **friction law** `<fric>`

- case 1 to use the Manning law; in this case, the **friction coefficient** `<friccoef>` [m$^{-1/3}$s] is Manning friction coefficient.
- case 2 to choose Darcy-Weisbach law; for this choice, the **friction coefficient** `<friccoef>` [dimensionless] is Darcy-Weisbach friction coefficient.

If you do not want to have friction, set the **friction coefficient** `<friccoef>` to 0.

3.1.4 Numerical scheme

The next six parameters are related to the numerical scheme. You have to choose the:

Numerical flux `<flux>`. You can use either:

- case 1: Rusanov flux.
- case 2: HLL (Harten,Lax and Van Leer) flux.
- case 3: HLL2 flux (it is an another algorithm to HLL).

Order of the scheme`<order>`. You can use either:

- case 1: order 1.
- case 2: order 2.

Linear reconstruction `<rec>`. This parameter will play a role only for a scheme of order 2.

- case 1: MUSCL.
- case 2: ENO.
- case 3: modified ENO.

Slope limiter `<lim>`. You can use either:

- case 1: the classical minmod slope limiter.
- case 2: the more complicated expression of Van Albada.
- case 3: Van Leer's reconstruction.

`<armortENO>`. This parameter must be specified when you are using linear reconstructions ENO or modified ENO. Its value must be between 0 and 1 (usually set to 0.25). If you take 0, it will be equivalent to choose the MUSCL reconstruction; if `<armortENO>` is set equal to 1, the reconstruction will be exactly ENO or modified ENO depending on your previous choice.

`<modifENO>`. For the modified ENO reconstruction, you can set the `<modifENO>` parameter between 0 and 1, usually taken equal to 0.9.

3.1.5 Infiltration

In this version of *FullSWOF_2D*, a modified Green-Ampt is the only infiltration model (see [7]). So, the parameter for the **choice of the infiltration model** `<inf>` can be equal to 1 only. This model assumes the soil is represented by two layers:

The first layer (top of the surface) represents a crust of **thickness** `<zcrust>` [m] and **hydraulic conductivity** `<Kc>` [m/s].

The second layer has a **saturated hydraulic conductivity** `<Ks>` [m/s].

For the two layers, the other parameters are:

- the **saturated moisture content** θ_s `<dtheta>` [dimensionless] between 0 (dry) and 1 (fully saturated).
- the **load pressure** ψ `<Psi>` [m].
- the **maximum infiltration rate** i_{max} `<imax>` [m/s]. In the standard Green-Ampt model, at $t = 0$ s, if the cumulative volume of water infiltrated is zero, the infiltration capacity is infinite. To avoid this phenomena, the Green-Ampt model has been modified to be able to impose a maximum infiltration rate i_{max}.

If you do not want to have infiltration, set `<Kc>` = 0.

3.1.6 Topography

The type of input is defined by `<topo>` which can take the following values:

- 1 to load the topography from a file. You have to enter the **topography** using an ASCII file you previously generated in the format "x y z". Do not forget to write the **name of your topography file** in `<topo_NF>`. This file must be in the *Inputs* directory.
- 2 to use a flat topography with $z = 0$;
- 3 to have the paraboloid defined by $z = -0.1*(1-\sqrt{(x-x_m)^2+(y-y_m)^2})$ where (x_m, y_m) is the center point of the computational domain (see [9]).

Remark 1. FullSWOF_2D *considers values (such as h, u, v, z) constant on a given cell, and the constant is given in the center of the cell, see figures 1 and 2.*

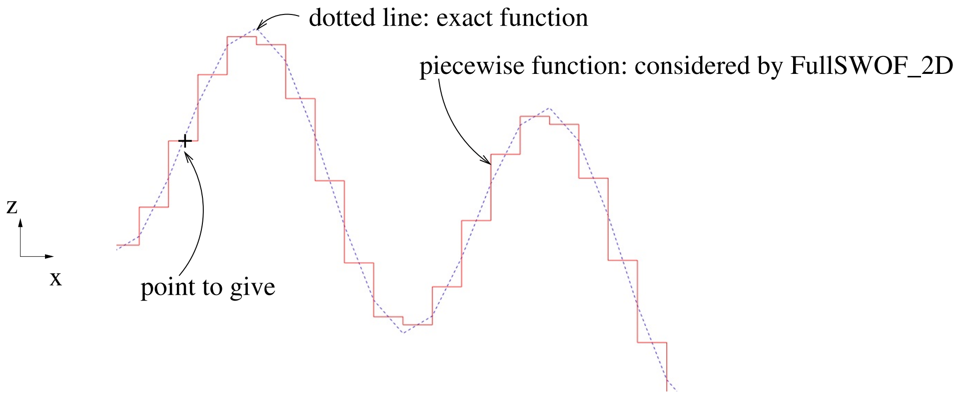

Figure 1: Piecewise approximation of 2D curves in *FullSWOF_2D*

3.1.7 Initial water height and velocity

You can impose the initial water height and velocity using `<huv_init>`:

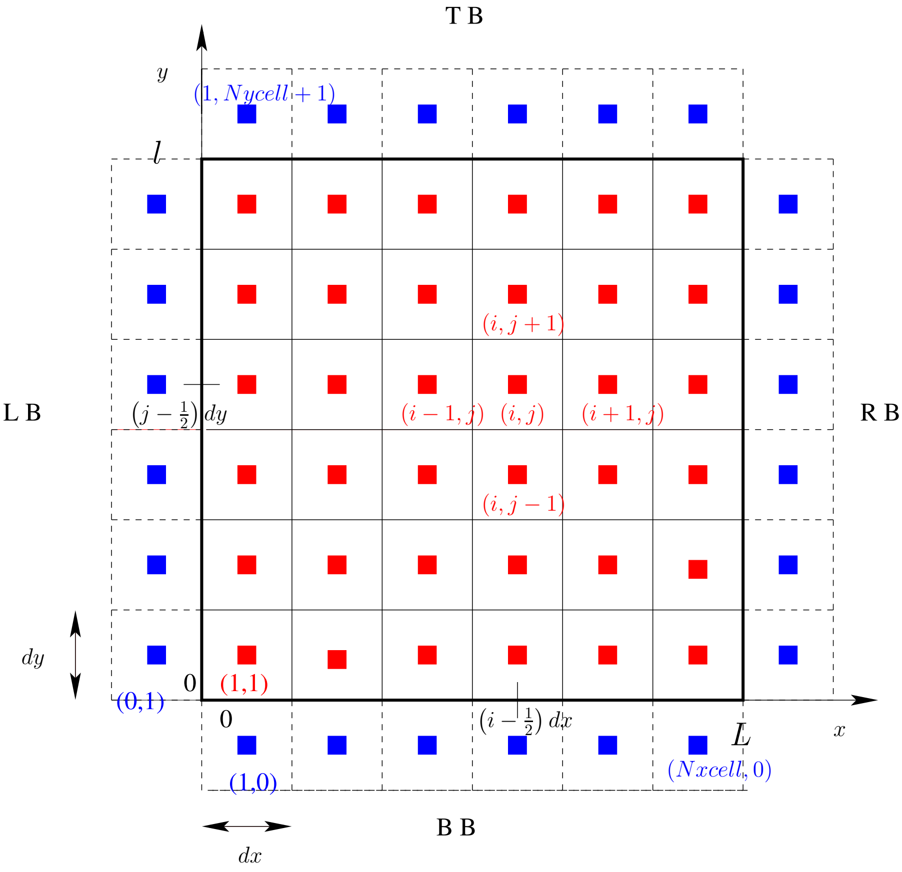

Figure 2: Mesh *FullSWOF_2D*;
LB= Left Boundary, RB= Right Boundary, BB= Bottom Boundary, TB= Top Boundary.
The computional domain is represented by the cells with the red center and the boundary corresponds to the cells with blue center.

- case 1 to load the initialisation of variables h [m], u [m/s] (velocity in the x-direction) and v [m/s] (velocity in the y-direction) from a file. This file must be in ASCII and follow the format "x y h u v". The **name of the file** `<huv_NF>` should be specified and this file must be in the *Inputs* directory.
- case 2 to have $h = 0$ m, $u = 0$ m/s and $v = 0$ m/s.
- case 3 to have the initialisation for the paraboloid used in [9].
  You should choose the Thacker topography for this case.
- case 4 to have $h = 0.005$ m in a disk centered at $(L/2, l/2)$ (which is the middle of the computational domain $[0; L] \times [0; l]$) with a radius $L/10$. $h = 0$ m outside the disk, and $u = v = 0$ m/s on the whole domain. This case is used to simulate a dry radial dam (see Figure 3, ref [8] and [1]).
  You should choose the flat topography for this case.
- case 5 to have $h = 0.005$ m in a disk centered at $(L/2, l/2)$ (which is the middle of the computational domain $[0; L] \times [0; l]$) with a radius $L/10$. $h = 0.001$ m outside the disk, and $u = v = 0$ m/s on the whole domain.This case is used to simulate a wet radial dam (see Figure 4 , ref [8] and [1]).
  You should choose the flat topography for this case.

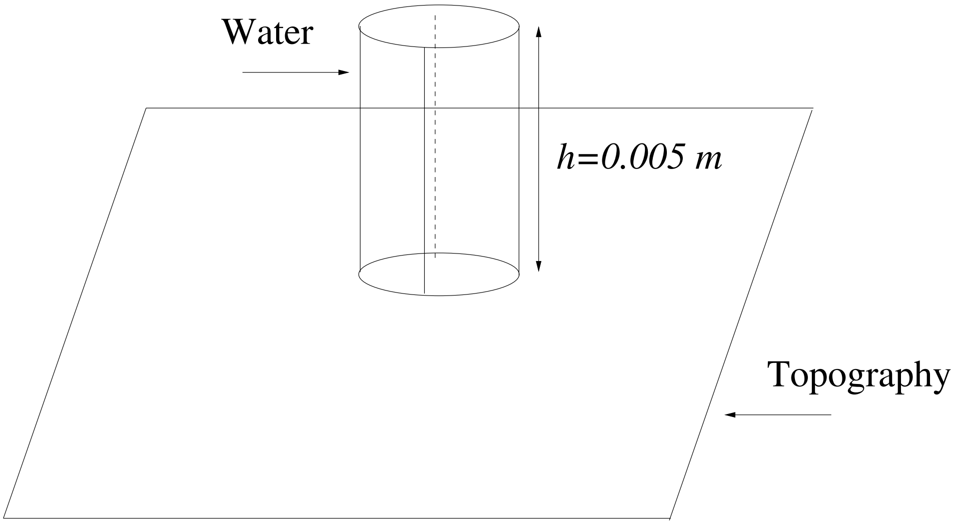

Figure 3: Initialization of h, u and v to simulate a dry radial dam: $h = 0$ m or $h = 0.005$ m, $u = 0$ m/s, $v = 0$ m/s.

3.1.8 Rain

For the **rain** `<rain>`, you can use:

- case 1: An ASCII file. The **name of your rain file** should be set for `<rain_NF>` and this file must be in the *Inputs* directory.
  The file must contain two columns:

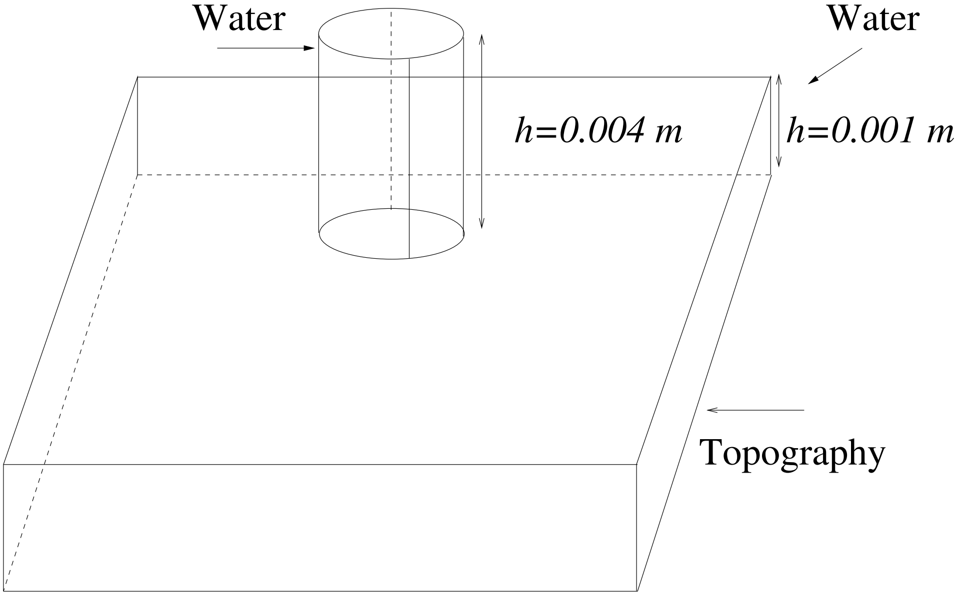

Figure 4: Initialization of h, u and v to simulate a wet radial dam: $h = 0.001$ m or $h = 0.005$ m, $u = 0$ m/s, $v = 0$ m/s.

  - the first one represents the time [s].
  - the second one the rain intensity [m/s].

  Each column must contain at least one value.
  In the first column, the first value must be 0 (initial time). If the file contains one value by column (namely 0 in the first column and any value in the second column), the rain intensity is constant during the run.
  If the file contains different values in each column, that means the rain intensity will be changed during the simulation.
  For example, let us consider three values for the time $(0, t_1, t_2)$ and the corresponding three values for the rain intensity (a_0, a_1, a_2) (written into the second column). Let us consider $Rain(t)$ the intensity of the rain at the time t, we will get the values:

$$Rain(t) = \begin{cases} a_0 & \text{if } 0 \leq t < t_1 \\ a_1 & \text{if } t_1 \leq t < t_2 \\ a_2 & \text{if } t_2 \leq t \end{cases}$$

  Be careful: the intensity may not change exactly at $t = t_1$ but at $t > t_1$. Indeed, *FullSWOF_2D* works with a constant CFL and the time step dt may not reach t_1 exactly. So, we can have $t < t_1$ for a given computing time, and the next computing may be $t + dt > t_1$. So, $t = t_1$ may not exist and the change in rain intensity will take place at $t + dt > t_1$.

- case 2: **No Rain**.

Remark 2. *You can create another case from, for example, the function in the rain_generated.cpp file (see the directory: Sources/librain_infiltration/).*

3.1.9 Name of output directory

The default name of the output directory is *Outputs*. However, you can add a **suffix** to this name (`<suffix_o>`). This is especially useful if you are running several tests.

3.1.10 Format of the output file

You have the choice between two possibilities to save the evolution of the computed values:

- case 1: gnuplot format is designed for the gnuplot software. This ASCII file can be used to draw easily the output using gnuplot software.
- case 2: vtk format can be used for the display of the computed values with paraview software. It is an ASCII file.

3.1.11 Comments

You can also add comments after the input value of each parameter. For example:

```
Time of simulation <T>:: 0.001 # try 0.01 next time
```

3.1.12 Advised values

For overland flow, several tests have been performed, in particular on the numerical scheme [3]. Consequently, we advise the user to choose the following numerical parameters:

- Numerical flux (1=Rus 2=HLL 3=HLL2) <flux>:: 2
- Order of the scheme <order>:: 2
- Reconstruction (1=MUSCL 2=ENO 3=ENOmod) <rec>:: 1
- Limiter (1=Minmod 2=VanAlbada 3=VanLeer) <lim>:: 1
- Cflfix, value of the cfl <cflfix>:: 0.4

If you choose the ENO or the modified ENO reconstruction, you should use:

- AmortENO <amortENO>:: 0.25
- ModifENO <modifENO>:: 0.9

3.2 *Outputs* directory

The results are saved in the *Outputs* directory. When starting the program, the following files are saved:

parameters.dat contains the parameters used by *FullSWOF_2D*, under the same form as the input file.

init.dat contains the initial conditions (water height, velocity, topography and free surface).

During the computation, some files are modified in order to save several time steps:

huv_movie.dat contains the evolution in time of the main variables (water height, velocity, free surface, topography, the Euclidean norm of the velocity, Froud number per cell, discharge along x, discharge along y, the Euclidean norm of the discharge).

boundaries_flux.dat contains the cumulated volume [m^3] output at each boundary.

flux_boundaries_LR.dat contains the flux [m^2/s] calculated at each cell of the left and right boundaries. .

flux_boundaries_BT.dat contains the flux [m^2/s] calculated at each cell of the bottom and top boundaries.

When the computation is done, three other files are created:

huv_final.dat contains the main variables (water height, velocity, free surface, topography, the Euclidean norm of the velocity, Froud number per cell, discharge along x, discharge along y, the Euclidean norm of the discharge) at the final time.

check_vol.dat contains the cumulated flux [m^3] at each time step. The first column is the time and the others are the cumulated fluxes for: overland flow volume (Vol_of_tot), infiltrated volume (Vol_inf_tot), rain (Vol_rain_tot) and the sum of the flux at the boundaries (bound_flux_tot).

Results.dat contains a mass balance to verify the scheme conservation at the final time. So, you can have:

- Infiltrated volume [m^3] is the total infitrated water volum cumulated during the simulation.
- Streamed volume [m^3] is the total water volume staying at the final time above topography.
- Complete volume (Inf+Stream) [m^3] is the sum of infiltrated volume and streamed volume.
- Volume of the rain [m^3] is the total rain volume cumulated during the simulation.
- Outflow volume [m^3] is the sum of the cumulated outflow volume at all boundaries.
- Duration of the computation [s].
- Number of iterations in the algorithm.
- Mean Froude number [dimensionless] in space at the final time.

4 For developers: Doxygen

You may wish to add some functionalities to *FullSWOF_2D* to suits your needs. Always comment the files, at the beginning of the file, using Doxygen syntax (`www.doxygen.org/`). Then, you will be able to create the doxygen documentation of the whole code.

HTML documentation. In order to generate the Doxygen html file, the **Doxygen_config_file_html** file is saved in the *doc* directory. To run Doxygen, from the *FullSWOF_2D* directory, use the command:

```
doxygen doc/Doxygen_config_file_html
```

Warning: Graphviz (`http://www.graphviz.org/`) must be in your PATH to generate HTML diagrams. If not, change the HAVE_DOT parameter of the Doxygen_config_file_html file. In the *doc/html/* directory, **index.html** is created.

PDF documentation. To generate the Doxygen LaTeX (pdf) file, you must use the **Doxygen_config_file_latex** file and compile the .tex file:

```
doxygen doc/Doxygen_config_file_latex
cd doc/latex
make
```

In the *doc/latex* directory, **refman.pdf** is created.

5 Validation

This software has been validated using several analytic solutions and benchmarks from the literature, gathered in [6]and in SWASHES `https://sourcesup.cru.fr/projects/swashes/`. In these documents, we recalled the main characteristics of these tests, and give the results of *FullSWOF_2D*.

Remark 3. *If you want to develop your own software in* FullSWOF_2D*, we advise you to run the benchmarks when installing the program, running* `make benchref`*. This will give you the reference solutions for each test. When you will get your modified code, just run all the test cases with* `make bench` *to be able to compare your results, in the Outputs directory, with the reference solutions, in the Outputs_REFERENCES directory. The analytic solution are also saved for each benchmark.*

In this documentation, we recall the main characteristics of these tests, and give the results of *FullSWOF_2D*.

5.1 Steady state solutions

In this section, we focus on a family of steady state solutions, that is solutions that satisfy:

$$\partial_t h = \partial_t u = \partial_t v = 0.$$

5.1.1 Bump

Here we present a steady state case (see [8]) with a flat topography at the boundaries, no rain, no friction and no diffusion. Initial conditions satisfy the hydrostatic equilibrium

$$h + z = Cst \text{ and } q = 0 \text{ m}^2/\text{s}. \tag{1}$$

This solution tests the preservation of steady state and the boundary conditions treatment.

In the following case, we choose a domain of length $L = 25$ m with a topography given by:

$$z(x, y) = max(0\ , 0.2 - 0.05 * (x - 10)^2 - 0.1 * (y - 10)^2)$$

Emerged bump at rest In the case of a lake at rest with an emerged bump, the water height is smaller than the topography heigth in order to have emergence of some parts of the bump. In such a configuration, starting from the steady state, the velocity must be null and the water surface should stay flat (see Figure 5).

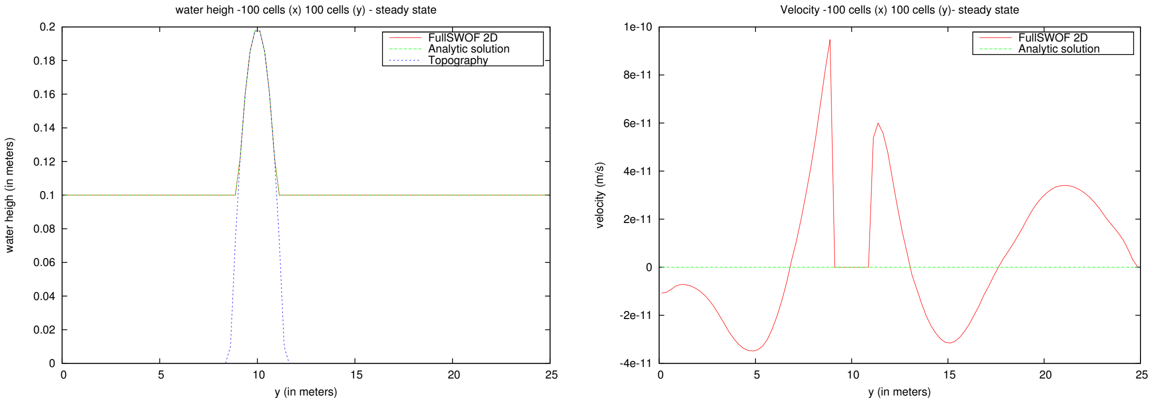

Figure 5: Emerged bump at rest

5.1.2 Fluvial MacDonald long channel with rain and Darcy-Weisbach friction coefficient

For a 1000 m long channel, we consider a flow which is fluvial on the whole domain (see Figure 6). Thus we impose the following boundary conditions:

$$\begin{cases} \text{upstream: } q = q_0, \\ \text{downstream: } h = h_{ex}(1000), \end{cases}$$

with the initial conditions

$$h = 0 \text{ m and } q = 0 \text{ m}^2/\text{s}.$$

For the friction term, we have chosen Darcy-Weisbach's law with $f = 0.093$, the discharge q_0 is set at 1 m^2/s and the rain intensity is $R_0 = 0.001$ m/s.

5.1.3 Torrential MacDonald long channel with rain and Darcy-Weisbach friction coefficient

The channel length remains unchanged (1000 m), but, as the flow is supercritical, the boundary conditions are

$$\begin{cases} \text{upstream: } q = q_0 \text{ and } h = h_{ex}(0), \\ \text{downstream: free.} \end{cases}$$

At initial time, the channel is dry

$$h = 0 \text{ m and } q = 0 \text{ m}^2/\text{s}.$$

The general form of the used rainfall event is

$$R(t) = \begin{cases} 0 \text{ m/s} & \text{if } t < t_R, \\ R_0 & \text{else,} \end{cases}$$

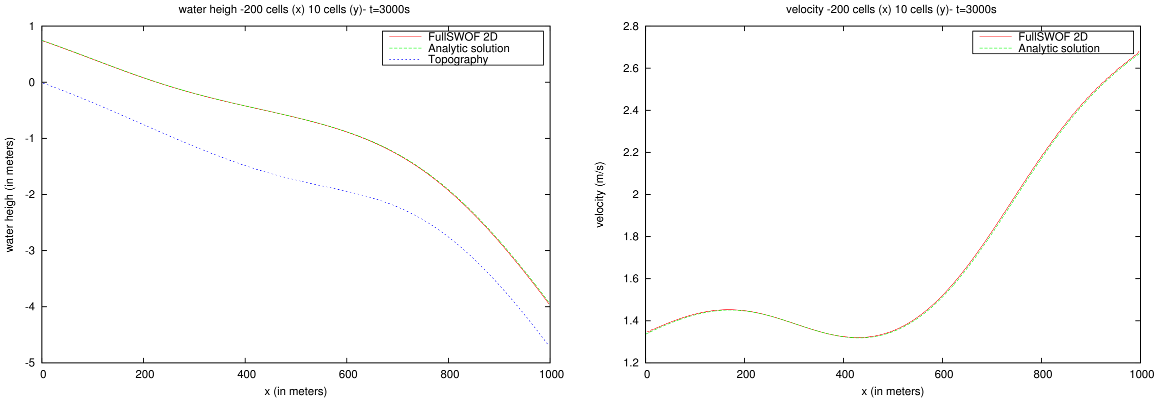

Figure 6: Fluvial MacDonald Rain long channel with Darcy-Weisbach friction coefficient

with $t_R = 1500$ s.

We have a friction coefficient $f = 0.065$ for Darcy-Weisbach's law. Inflow discharge is $q_0 = 2.5$ m^2/s and $R_0 = 0.001$ m/s (see Figure 7).

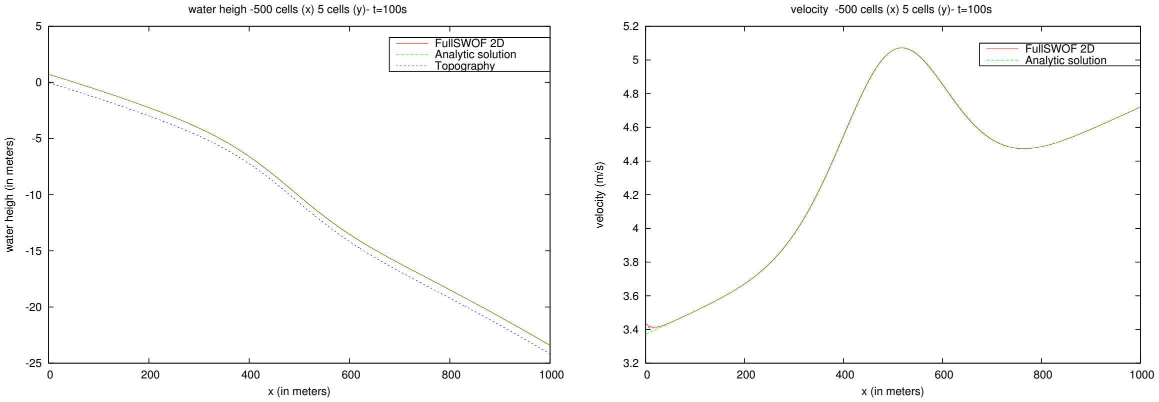

Figure 7: Torrential MacDonald: Rain long channel with Darcy-Weisbach friction coefficient

5.1.4 MacDonald short channel with smooth transition and shock

The length of the channel is 100 m and the discharge at steady state is $q = 2$ m^2/s. The flow is fluvial both upstream and downstream. In this case, the Manning's friction coefficient is $n = 0.0328$ m$^{-1/3}$s, the inflow is subcritical, becomes supercritical via a sonic point, and, through a shock (located at $x = 200/3 \approx 66.67$ m), becomes subcritical again (see Figure 8).

5.2 Mac Donald pseudo-2D solutions

In this section, we give several analytic solutions for the pseudo-2D Shallow-Water system. This system can be considered as an intermediate between the one-dimensional and the two-dimensional models. More precisely, these equations model a flow in a rectilinear three-dimensional

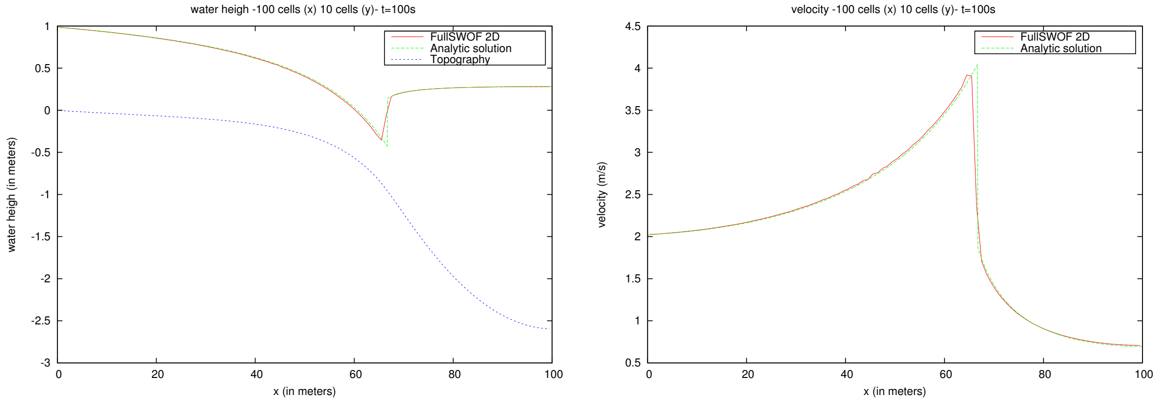

Figure 8: MacDonald short channel smooth transition shock

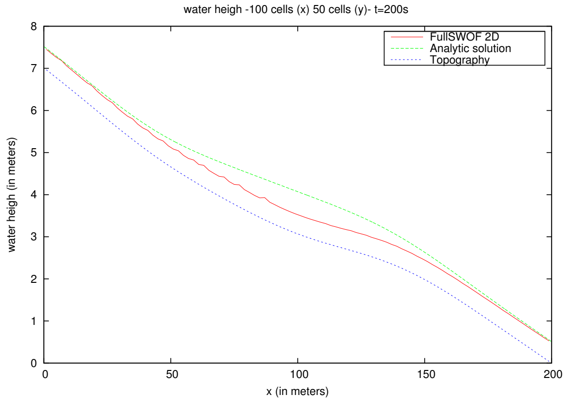

Figure 9: Torrential Mac Donald pseudo 2D short channel with Manning friction coefficient

channel with the quantities averaged not only on the vertical direction but also on the width of the channel.

We consider two cases for non-prismatic channels. Each channel is determined through the definition of the bottom width B (as a function of the space variable x, see **SWASHES**) and the slope of the boundary Z. The bed slope is an explicit function of the water height.

Torrential Mac Donald pseudo-2D short channel with Manning friction coefficient In this case, the flow and the water height are set at inflow (see Figure 9). We have the following boundary conditions:

$$\begin{cases} \text{upstream: } q = 20 \text{ m}^3\text{s}^{-1} \text{ and } h = h_{in}, \\ \text{downstream: free.} \end{cases}$$

The channel is initially dry, *i.e.* initial conditions are:

$$h = 0 \text{ m}, \text{ and } q = 0 \text{ m}^3/\text{s}.$$

In this case, the flow is supercritical.

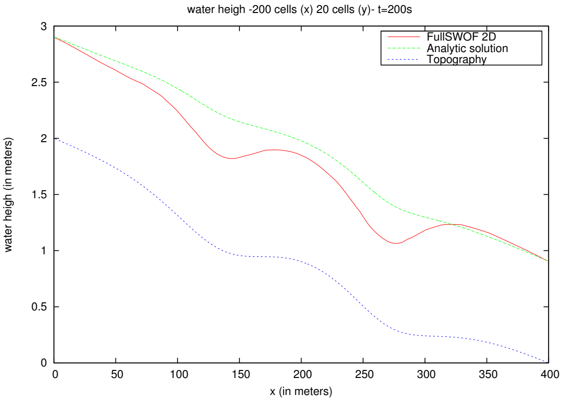

Figure 10: Fluvial Mac Donald pseudo 2D long channel with Manning friction coefficient

Fluvial Mac Donald pseudo-2D long channel with Manning friction coefficient From now on, the length of the domain is $L = 400$ m, the boundaries of the channel are given by B_2 and the cross sections are isoscele trapezoids.

In this case, the flow is set at the inflow and the water height is prescribed at the outflow (see Figure 10). We have the following boundary conditions:

$$\begin{cases} \text{upstream: } q = 20 \text{ m}^3\text{s}^{-1}, \\ \text{downstream: } h = h_{out}. \end{cases}$$

The channel is initially dry, with a little puddle downstream. In this case, the flow is subcritical along the whole channel.

5.3 Transitory solutions

In the previous section, we gave steady-state solutions of increasing difficulties. These solutions can be used to check if the numerical methods are able to keep/catch steady-state flows. But even if the initial condition differs from the expected steady state, we do not have information about the transitory behavior. Thus, in this section, we give some transitory solutions that may improve the validation of the numerical methods. Moreover, as these cases have wet/dry transitions, one can check the ability of the schemes to capture the evolution of these fronts (*e.g.* some methods may fail and give negative water height). At last, we give a periodic transitory solutions in order to check whether the schemes are numerically diffusive or not.

5.3.1 Dam break on a dry domain without friction

In this section, we are interested in a dam break solution on a flat topography namely Ritter's solution. The initial condition for this configuration is the following Riemann problem

$$h(x,y) = \begin{cases} 0.005 & \text{for } 0 \text{ m} \leq y \leq L - x, \\ 0 & \text{for } y > L - x, \end{cases}$$

with $u(x,y) = v(x,y) = 0$ m/s.

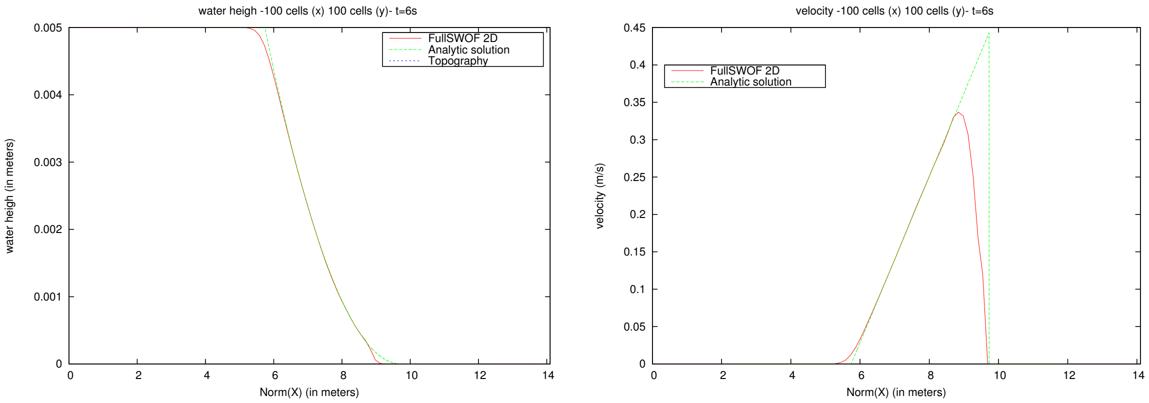

Figure 11: Dam break dry without friction

5.3.2 Thacker test case with planar surface in paroboloid

For this Thacker's 2D case, the moving shoreline is a circle and the topography is is a paraboloid of revolution. The free surface has a periodic motion and remains planar in time [9]. To visualize this case, one can think of a glass with some liquid in rotation inside.

Here again, the analytic solution at $t = 0$ s is taken as initial condition (see Figure 12).

The topography

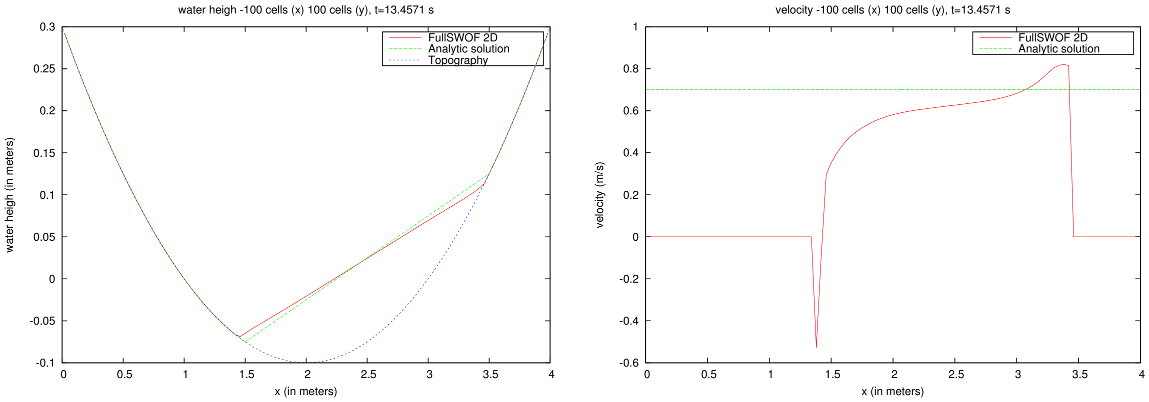

Figure 12: Thacker test case with planar surface in paroboloid

References

[1] Emmanuel Audusse. Schémas cinétiques pour le système de saint-venant. Master's thesis, Université Paris 6, 1999.

[2] Olivier Delestre. Ecriture d'un code C++ pour la simulation en hydrologie. Master's thesis, Université d'Orléans, Orléans, France, September 2008.

[3] Olivier Delestre. *Simulation du ruissellement d'eau de pluie sur des surfaces agricoles*. PhD thesis, Université d'Orléans, Orléans, France, July 2010.

[4] Olivier Delestre, Stéphane Cordier, François James, and Frédéric Darboux. Simulation of rain-water overland-flow. In E. Tadmor, J.-G. Liu, and A. Tzavaras, editors, *Proceedings of the* 12th *International Conference on Hyperbolic Problems*, volume 67 of *Proceedings of Symposia in Applied Mathematics*, pages 537–546, University of Maryland, College Park (USA),, 2009. Amer. Math. Soc.

[5] Olivier Delestre and François James. Simulation of rainfall events and overland flow. In *Proceedings of X International Conference Zaragoza-Pau on Applied Mathematics and Statistics, Jaca, Spain*, Monografías Matemáticas García de Galdeano, 2009.

[6] Olivier Delestre, Carine Lucas, Pierre-Antoine Ksinant, Frédéric Darboux, Christian Laguerre, Thi Ngoc Tuoi Vo, François James, and Stéphane Cordier. SWASH: a library of Shallow-Water Analytic Solutions for Hydraulics. To be submitted.

[7] Michel Esteves, Xavier Faucher, Sylvie Galle, and Michel Vauclin. Overland flow and infiltration modelling for small plots during unsteady rain: numerical results versus observed values. *Journal of Hydrology*, 228:265–282, 2000.

[8] Nicole Goutal and François Maurel. Proceedings of the 2nd workshop on dam-break wave simulation. Technical Report HE-43/97/016/B, Electricité de France, Direction des études et recherches, 1997.

[9] William Carlisle Thacker. Some exact solutions to the nonlinear shallow-water wave equations. *Journal of Fluid Mechanics*, 107:499–508, 1981.