Reference

- supports parallel processing
- with dynamic memory allocation

Content

Modules

`debris flow`	non-newtonian fluids
`sediment`	mobile bed calculation
`concentration`	advective transport (pollution, thermal inlets)

1 Reference

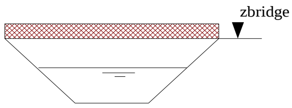
Input	*Unit*	*Description*
`title 'name'`	string	**'name'** is a string (max. 64 characters) that is stored on the result file and appears in the header of the plots
`>>global`		main keyword for the definition of default values holding for all models
`kst r`	$m^{1/3}/s$	global Strickler value (default=30)
`n r`	SI	default Manning's n value
`ks r`	m	equivalent sand roughness diameter
`damping_on`		considers reduction of turbulent shear forces for small flow depths (Bezzola 2002) in combination with logaritmic friction law (sand roughness ks).
`label 'xx' r`	string	name of a label (max. 4 characters) to which the Strickler-value r is related
`hdry r`	m	minimum flow depth where flow equations are being solved (for 2D model, default=0.01 m)
`adry r`	m^2	minimum wetted area of cross-section where flow equations are being solved (for 1D model, default=0.1 m^2)
`slot_width r`	m	width of (Preissmann-) slot used for calculation of pressurised flows (see option >>branch/closed) (default=0.1 m)
`unit 'name'`	string	Defines unit of input data. Possible units are: - 'minutes' for time given in minutes (instead of hour)
`date 'mmddhh'`	string	real time of start of simulation (mm=month, dd=day, hh=hour)
`yield_stress`	N/m^2	yield stress for debris flow calculations (default = 0)
`friction_slope`	-	Tangens of internal friction angle $\tan\delta$ for debris flow calculations (default = 0) with viscous stress $\tau_y = \rho\, g\, h \tan\delta$. <u>Note</u>: If both yield stress and friction slope are defined, the sum of the two values is taken.
`bingham_viscosity`	Pa s	Bingham viscosity μ_B for debris flow calculations (default = 0) with Bingham shear stress $\tau_B = \frac{3\mu_B q}{h^2}$. Note: If both turbulent friction (n, kst, or ks) and bingham viscosity is defined, the maximum of the two resulting stress values is taken.
`roughness_factor`	-	factor to estimate roughness diameter from mean grain-size
		parameters for integration of 1D models
`seam_radius r`	m	maximum distance from section-midpoints to cell-boundaries (of 2D mesh) where a seam (flow) can exist (used for connecting different flow models) (default = 10 m)
`weir_coefficient`	-	Poleni coefficient for calculation of fluxes between 1D and 2D models (default = 0.60)
`concentration 'name'`	string	activates advective transport simulation
`>>sediment`		*parameters for mobile bed calculation*

Input	***Unit***	***Description***
`thcrit r`	-	critical shields factor for MPM and Smart/Jäggi formula (default=0.05)
`repose r`	-	tangens of angle of repose of bank material (default=1.0)
`density r`	kg/m^3	density of the bed material (default=2650 kg/m^3)
`porosity r`	-	porosity of the bed material (default=0.30)
`formula 'name'`	-	sediment transport formula to be used: • **`mpm`** = Meyer-Peter/Müller formula. • **`smart&jaeggi`** = Smart/Jäggi formula • **`parker78`** = Parker formula for uniform grain size
`rock_thcrit r`	-	critical shields factor for transport over bedrock (default=0.01)
`rock_factor r`	-	factor to account for transport over bedrock (default=1.8)
`mpm_factor r`	-	factor used in transport formula of Meyer-Peter/Müller (default=8.0)

`mixture 'name'`
`0.2 0.`
`. .`
`. .`
`25. 1.`
	cm	grain size distribution of sediment mixture where the the grain size [cm] and the cumulative probability (sediment finer) are given in the 1st and 2nd column.
		Note: The last value in the 1st column must be 1.0.

`>>compute`		to define parameters for unsteady flow computation
`start r`	h	start time of the simulation (default=0)
`end r`	h	time where simulation will end (default=100h)
`cfl r`	-	limiting CFL number to estimate size of time step (default=0.6)
`frequency i`	-	refresh rate of display output (default=100).
`batch_mode`		runs the model in batch_mode, i.e. starts the computation, stores the results on the specified file, and terminates
`plot_interval`	h	interval to store sediment output on file sed.out (default= 8760)
`num_threads i`	-	Maximum number of threads for computation (default = 1). Speeds up calculation on systems with multi-core (multi-thread) CPU's.
`>>create_model`		main keyword for the specification of model-specific parameters
`name 'name'`	string	name of model (displayed on model output)
`type '2D'`		type of model: • '1D' for one-dimensional flow calculations (river branch) • '2D' for two-dimensional flow calculations
`>>init`		definition of the calculation domain
`mesh 'name'`	string	reads the mesh geometry from files. Supported formats are: ***.node*** created by program TRIANGLE (file name without suffix!) ***.2dm*** created by program SMS (splits 4-noded elements to triangles) ***.tin*** triangulated irregular network format (used in ems-i programs)

Input	***Unit***	***Description***
`binary ´name´`	string	reads mesh geometry and initial conditions (flow depths, flow and bedlevels) from a binary file created by a previous run. *Note: Use either* ***`mesh`*** *or* ***`binary`*** *(not both) for the definition of the mesh geometry.*
`at r`	h	time level r of initial condition to be read form the binary file.
`bedlevel =+-<> 'name'`		reads a mesh geometry from files created by program triangle. Depending on the operator it changes the level of the model bed. Example: bedlevel = 'new_dam.1' will read the bedlevel values from the mesh defined by the files new_dam.1.node and new_dam.1.ele. Possible operators are: = new bedlevel + lift the bedlevel - lower the bedlevel < maximum bedlevel > minimum bedlevel *Note: Must be used in combination with mesh (not binary).*
`waterlevel =+-<> 'name'`		Same as option bedlevel (see row above) but operates on waterlevels. *Note: Must be defined after definition of bedlevels.*
`>>polygon`		used to define values that hold in a domain defined by a closed polygon. Example: **`>>polygon          !keyword`** **`  n 0.02           !keyword for Manning's n value`** **`  100. 150.        !list of vertices of polygon`** **`  320. 165.`** **`  240. 190.`** **`  105. 155.`**
`bedlevel =+-<> r`	m	to modify the bedlevels by a value r using an operator (see keyword `>>init/bedlevel` for meaning of operators). Example: **`bedlevel > 321.0 ! minimum bed level is 321 m`** **`  100. 150.`** **`  320. 165.`** **`  240. 190.`** **`  105. 155.`**
`flowdepth r`	m	flow depth at start time (initial condition)
`ks r`	m	equivalent sand roughness diameter
`kst r`	SI	Manning-Strickler value
`n r`	SI	Manning's n value
`vegetation r`	1/m	vegetation factor given by the formula $vegetation = \frac{d}{a^2} c_w$ with d = diameter of vegetation elements [m], a = distance between elements [m] and c_w = drag coefficient (range 0.8 – 1.5). Can be used to account for drift wood effects (see >>drift).

Input	***Unit***	***Description***
`waterlevel r`	m	water level at start time (initial condition)
`concentration r`	-	concentration at start time (initial condition; default = 0)
`bridge r`	m	level of a bridge (z_{bridge}) to account for backwater effects. It accounts for the acceleration of the flow due to the reduced flow section. It does not account for external forces on the bridge plate or other effects such as flow contraction (gated flows). 

`no_seam`
`x1 y1`
`x2 y2`
`.  .`
	-	closed polygon to define cells that do not connect to 1D-branches.

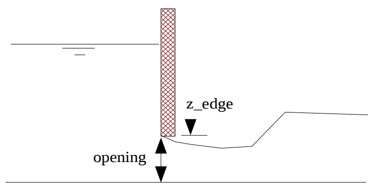
`>>boundary`		used to define (time-dependent) boundary conditions at the model boundaries that are inside a polygon. The polygon covers all the edges of the calculation mesh where the boundary condition holds. It must not match exactly with the edges. The steps are: (1) Define a boundary type (e.g. an inflow) (2) Define the location where the boundary holds by a closed polygon using the keyword **`location`**.
`inflow r` `uniform_slope r`	m^3/s -	defines an inflow discharge (** for time dependent inflow). The discharge is distributed among the boundary cells assuming uniform flow conditions given the slope of the energy head (uniform_slope, default = 0.001).
`critical`		defines an outflow boundary with a critical flow regime (no backwater effect)
`slope r`	-	defines an outflow boundary with r = energy slope
`waterlevel r`	m	defines an outflow boundary with r = water level

`stage-discharge`
` z1  q1`
` z2  q2`
` .   .`
		defines an stage-discharge outflow boundary with z=water level [m] and q=discharge [m^3/s]

Example (i) Given an inflow increasing from 5 to 100 m^3/s within half an hour at a boundary where the mean slope is approx. 0.5%. The boundary condition reads

```
>>boundary
  inflow **
  0.  5.
  0.5 100.
  uniform_slope 0.005
  location
    100. 150.
    100. 150.
    200. 200.
    100. 200.
```

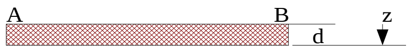
Input	*Unit*	*Description*
		Example (ii) At an outflow boundary the water level rises from 96.5 m to 98.0 m during half an hour and returns to the old value after one hour. The outflow has to be stored on the file 'waterlevel.out' for further usage. The boundary condition reads **`>>boundary`** **`  waterlevel ** > 'waterlevel.out'`** **`    0.0 96.5`** **`    0.5 98.0`** **`    1.0 96.5`** **`  location`** **`    900. 150.`** **`    930. 155.`** **`    950. 240.`** **`    910. 260.`**
`concentration r`	[-]	concentration of the inflow (defined above)
`>>structure`		to define internal sources and structures (culverts, weirs, controls)
`culvert x1 y1 x2 y2`	m	defines the flow through a circular or rectangular culvert with (x1,y1) and (x2,y2) = co-ordinates of the in- and outlet. The module accounts for in- or outlet controlled flow condition. It is assumed that the vertical level of the in- and outlet corresponds to the bed level of the adjacent grid cell. Example: **`culvert x1 y1 x2 y2 > ´name´`** writes the discharge through the culvert to file ´name´
`diameter r`	m	diameter of circular culvert (default= 1 m)
`width r`	m	width of rectangular culvert
`height r`	m	height of rectangular culvert.
`n r`	SI	Manning´s n value of culvert [default= 0.02].
`kst r`	SI	Strickler value of culvert [default= 50].
`inlet_loss r`	-	inlet loss coefficient that depends on shape of culvert inlet. Values usually vary between 0.2 (rounded entrance) and 0.7 (sharp crested entrance)(default= 0.5).
`maximum  r`	m^3/s	maximum discharge through culvert (** for time table).

Input	Unit	Description
`weir zw cw`		flow over weir with zw = level of weir crest [m] and cw = poleni coefficient (default = 0.58). Flow over weir can be written to a file by adding > 'filename'. Example: **`weir 433.65 0.64 > 'weirflow.out'`** **`location`** `...` Time_dependent weir levels are defined in a table where the time [h] and the weir_crest are given in the first and second column, respectively. Example: **`weir ** 0.64`** **`0.0 433.65`** **`0.8 434.15`** **`1.5 433.80`** **`location`** `...`
`gate zg cg`	m,-	flow through gate with zg = level of the lower end of the sluice gate (see figure) and cg = contraction coefficient (default = 0.62). Flow through gate can be written to a file by adding > 'filename'. Example: **`gate 426.45 0.80 > 'gateflow.out'`** **`location`** `...` Time_dependent gate openings are defined in a table where the time [h] and the opening are given in the first and second column, respectively. Example: **`gate ** 0.80`** **`0.0 426.45`** **`0.1 425.00`** **`location`** `...` 
`control_gauge x y z`	m,m,m	water level control with reference level z [m] at position defined by co-ordinates x,y. Control values can be written to a file by adding > 'filename'.
`control_param dt u zm qm`		parameters of level control (fictitious weir): ● dt = time lag [s] between adjustment of weir level ● u = velocity [m/s] of weir level adjustment ● zm = minimum weir crest [m] (default=-9999.) ● qm = maximum weir flow [m3/s] (default=9999.)
`control_init zw`	m	initial level of (fictitious weir) crest (default: control level) Example: **`control_gauge 29.5 94.8 433.65 >'control.out'`** **`control_param 120. 0.002 431.0 500.`** **`control_init 433.20`** **`location`** `...`

Input	***Unit***	***Description***

`location`
`x1 y1`
`x2 y2`
`. .`
		location of weir/gate/control section defined as a set of vertices with x- and y- coordinates in first and second column, respectively.

`pile x y d cD`		accounts for drag forces on a pile (pier) at location x,y with d= diameter [m], cD = drag coefficient (default = 1.0 for circular shape).

`cross-pile z d cD`
`xA yA`
`xB yB`
		accounts for drag forces on a horizontal element that spans from A to B with z = level [m], d = diameter [m], cD = drag coefficient (default = 1.0 for circular shape).
		
		Hint: Cross-piles can be used to model drag forces due to bridge plates.

`>>variation`		is used to define variation of model parameters during simulation

`bedlevel =+-<> **`
`t1 r1`
`t2 r2`
`. .`
	m	definition of time-dependent values of bed level using an operator (see keyword `>>init/bedlevel` for meaning of operators).

`rock_level = **`
`t1 r1`
`t2 r2`
`. .`
	m	definition of time-dependent values of rock level (for bed load calculation).

`location` `. .`		co-ordinates of polygon vertices to define location of variation.
`deposit r1 r2 r3 r4 r5`		adds depth of flow to the bed level (to account for deposition of debris flow) at time `r1` [h]. Simultaneous the global (rheological) parameters can be changed: `kst (r2)`, `yield_stress (r3)`, `friction_slope (r4)`, and `bingham_viscosity (r5)`
`>>source`		internal sources or sinks
`discharge q`	m3/s	defines **`q`** = constant discharge. For unsteady inflows (hydrographs) write '**' and add a timetable of the inflows.
`concentration r`	[-]	concentration of the discharge (defined above)
`point x y`		location of point source with coordinates **`x,y`**

`polygon`
`x1 y1`
`x2 y2`
`. .`
		area (closed polygon) where the discharge (defined above) is equally distributed.

`>>sediment`		the following data is related to bed load calculations
`dmo r`	cm	reads mean diameter of grains (valid for whole domain)

Input	*Unit*	*Description*
`dmo 'name'`	string	reads variable grain size as a mesh created by program triangle ("name.node" and "name.ele"). To be used for uniform grain size calculation (unit [cm]).
`inflow r`	kg/s	sediment inflow (r=** for unsteady inflows).

`location`
`x1 y1`
`x2 y2`
`. .`
		location of sediment inflow (feeder) given as a closed polygon

`rock_depth r`
`x1 y1`
`x2 y2`
`. .`
	m	defines a rock level below the initial bed level in a closed polygon defined by the co-ordinates x,y

`>>output`		is used to define the output from the model
`prefix 'name'`	string	results are written to file **`name.res`** (binary for review) and `name.out` (input reading)
`interval r`	h	interval between time steps to be stored on result file (default = 1 h)

`hydrograph 'item' x y > 'filename'`

writes nodal value at location (x,y) as a hydrograph table to file. Accepted items are:

`'bedlevel'`	bed level [m s. l.]
`'waterlevel'`	water level [m s. l.]
`'depth'`	flow depth [m]
`'flow'`	Specific flow [m2/s]
`'velocity'`	flow velocity [m/s]

`write 'item' at r > 'filenname'`

write nodal values at time r [h] to file. See table above for accepted items.

`flow > 'fname'`
`x1 y1`
`x2 y2`
`. .`
	writes total flow across a section to file 'fname'. The section must be defined as a polygon with vertices x,y. The location of edges can be displayed with Show/Mesh option. Note that the direction of the flow is <u>not</u> considered..

`sediment > 'fname'`
`x1 y1`
`x2 y2`
`. .`
	writes sediment flow across a section to file 'fname'. The section must be defined as a polygon with vertices x,y. The location of edges can be displayed with Show/Mesh option. Note that the direction of the flow is <u>not</u> considered..

>> denotes the end of the input. Any further input is ignored.

2 History

Version 3.0 (2016-)

- unlimited model size (allocatable arrays)
- parallel processing (multi-thread CPU's)
- initial weir levels (for waterlevel controls)
- convective transport (e.g. concentration or temperature)

Version 2.3 (2014-15)

- >>variation for time-dependent rocklevels
- adjustment of stress terms for debris flow calculations

Version 2.2 (2011-2013)

- module to account for coarse woody debris
- bingham like friction law for debris flow calculations.
- >>variation for adjusting values (e.g. bedlevels) in time.

Version 2.1 (2009-2011)

- accepts mesh geometries in .2dm format created by SMS (Surface water Modeling System) and .tin format (see description)
- serial linking of 2d and 1d models
- cross-piles to account for hydraulic resistance of horizontal structures such as bridges
- export of results in (ESRI) shape format
- simplified input for flow over weirs and gated flows
- modeling of water level controls (e.g. hydro power stations)
- modeling debris flow with two-parameter approach (turbulent & yield)

Version 2.0 (2006-2009)

- improved integration of multiple models (1D and 2D)
- accepts project files to change river bed topography

Version 1.0 - 1.3 (1999-2005)

- stage-discharge outflow boundary
- development of breaches during simulation time
- export of hydrograph tables
- variable bed_evolution values and boundary conditions
- bed armouring (2-grain-size model)
- automatic generation of animated output (movies)
- perspective views with shading
- modelling of precipitation/evaporation
- energy losses due to vegetation
- multiple (independent) meshes
- mobile bed module
- improved interpolation of bed level for narrow dams
- improved graphic routines
- output of multiple time steps on same result file
- logarithmic friction law
- backwater effects due to bridges
- time dependent boundary conditions
- distributed inflow discharge assuming uniform flow conditions
- culvert flow to connect model domains
- weir and gated flow over cell edges
- friction values (kst or n) defined with closed polygons (mesh independent)
- wetting and drying of cells