

## WaSiM 8.01.02 „Whats New“

WaSiM 8.01.02 now includes two new components: a surface routing component and a silting up (or coating/crusting) component. The following chapters will describe the functional design and the theoretical basis as well as the additionally required input data and parameters and changes to the control file.

### Key features of both modules

- **Silting-up Module**

- infiltration into soil is controlled by an exponential recession function
- main parameters for infiltration rate are
  - the cumulated kinetic energy of a rain event,
  - the silting-up disposition which depends on soil type and time since last tillage
  - and the end infiltration rate depending on the soil type (fractions of silt, sand and clay)
- after a rain pause which is long enough for recreation of the soil, the cumulated kinetic energy is set to 0 again.
- Actual infiltration rate using the above mentioned parameters is compared to the actual precipitation rate → excessive water will be branched off as a surface runoff component
- remaining water is transferred to the unsaturated zone model as input
- there, infiltration excess or even exfiltration from groundwater will be added to the surface discharge from the silting up module.
- No parameter change for the soil model necessary – all additional parameters are soil type specific and only used in the silting up model
- Silting-up can be used land use specific: When tillage days are given in the land use table, then silting-up will be calculated, otherwise the silting-up-schema is skipped.

- **surface routing**

- designed for small scale applications, otherwise it's too physically based (applications are still possible, but the interpretation will become more difficult)
- infiltration excess of soil- and silting-up model and direct runoff from snow melt is used as input for the surface routing model
- surface runoff will follow the flow paths from cell to cell until a river cell is reached (note: at least one river cell is required in each subbasin - preferably at the lowest point - otherwise no surface runoff will be generated at all)
- in diverging areas, multiple flow paths (up to three) are possible (if plan curvature  $< 0$ , flow paths are diverging, otherwise converging) → use method 1 or 3
- in converging areas, only one flow direction is possible (following the steepest slope)
- flow velocity depends on slope, roughness and water film depth (Manning Strickler approach)
- runoff is assumed to flow as a consistent layer → however, a concentration factor may be used to take into account small scale variations of the surface leading to preferred flow paths in small channels
- slope takes into account the elevation model plus the water storage, thus a dynamic generation of ponds is possible (if elevation + local storage leads to a zero or inverted gradient) → it is still not a hydrodynamic wave approach but rather a cinematic wave approach with stepwise constant parameters. This option is available using methods 3 and 4, only. Methods 1 and 2 use constant slopes without regarding water levels
- roughness can be estimated by different approaches for different types of cultivation

- fix value for uncultivated areas (e.g. forest)
- land use dependent parameters for conserving cultivation
- dynamically estimated parameters for flow through vegetation (originally designed for non-conserving cultivation) taking into account vegetation specific parameters in their annual phenologic course and other hydraulic properties
- time step of surface routing is dynamically decreased, if necessary (can be parameterized)
- surface storage of non discharged water is input into the silting-up model or the soil model (depending of configuration) in the next time step
- interflow and base flow are passed through from unsaturated zone model (soil model)
- surface runoff and interflow can optionally be lead into a single reservoir cascade to model the translation and transforming effects of rivers within a subbasin → the same approach as used in the soil model when no surface routing is active. The difference between model run with and without surface runoff is, that for model runs with surface discharge routing only the discharge from cells with a river on it flows into the reservoir cascade. To disable the reservoir cascade, the cell values of the flow time grid may be set to 1 (second) to make sure that all runoff counts for the actual interval only (and the storage recession parameter must be set to a very short time like 0.0001 h).
- total runoff is calculated by adding the subbasin-wide averages of all three runoff components
- total runoff is then input into routing model
- tracer mixing is fully integrated into surface routing

## Functional design

Figure 1 delineates the previous schema for water flows (including tracers). If surface routing and silting-up are switched off, then the new WaSiM version 8.1.01 is functionally identical to the previous versions, so the schema is still valid in new versions for that case.

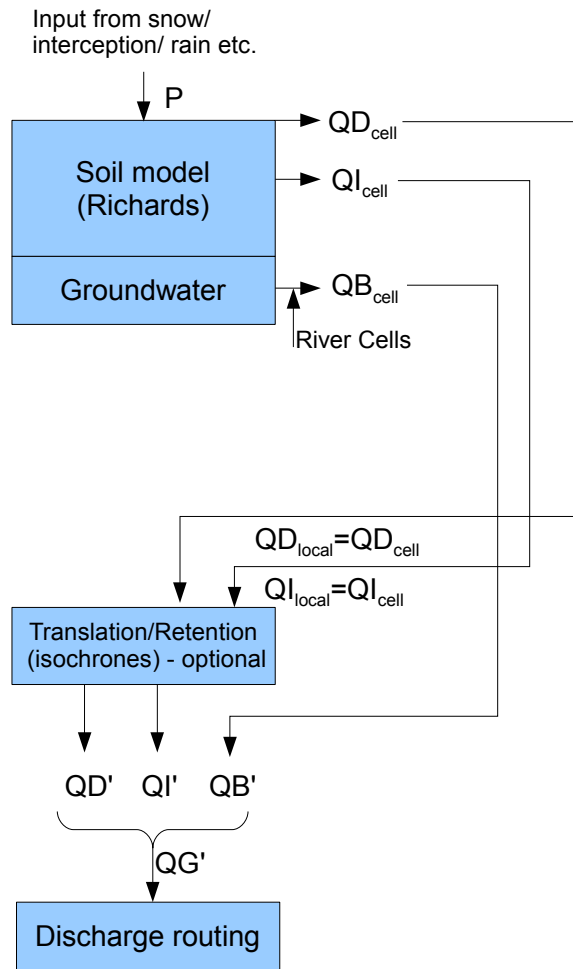


Figure 1: Flow chart for traditional soil- and routing modelling (prior to version 8.1.01 and without silting-up and surface routing since version 8.1.01)

As shown, the soil model generates basically three runoff components (drainage runoff is handled as part of interflow and thus not explicitly shown here): direct discharge (QD), interflow (QI), and base flow (QB). If a flow time grid was read in, then QD and QI are transformed separately by a single reservoir cascade (isochrone approach with additional retention in each class). The average values of each runoff component  $QB'$ ,  $QI'$ , and  $QD'$  are summed up to the total runoff  $QG'$  which is then the input to the routing model.

Now, starting with version 8.1.01, two new components may be used to model the generation and transformation of surface runoff. Figure 2 shows the flow chart for the complete schema. The silting-up module simply separates a fraction of the incoming water (usually precipitation) and stores it as direct discharge  $QD_{SU}$  (SU standing for Silting Up). The following Richards Schema is identical to the previous existing one, except for adding the direct discharge generated within the soil model to the already generated direct discharge from the silting-up model. Both components together are written as direct discharge statistics file or grid of the soil model. Note: when using

surface routing, the QD statistics file and grids will contain the surface storage for direct runoff, so QD from soil model cannot be used for balancing anymore → use  $QD_{SR}$  instead, the statistics output from SurfaceRouting model, or, if no surface routing is active, use QD from soil model as usual.

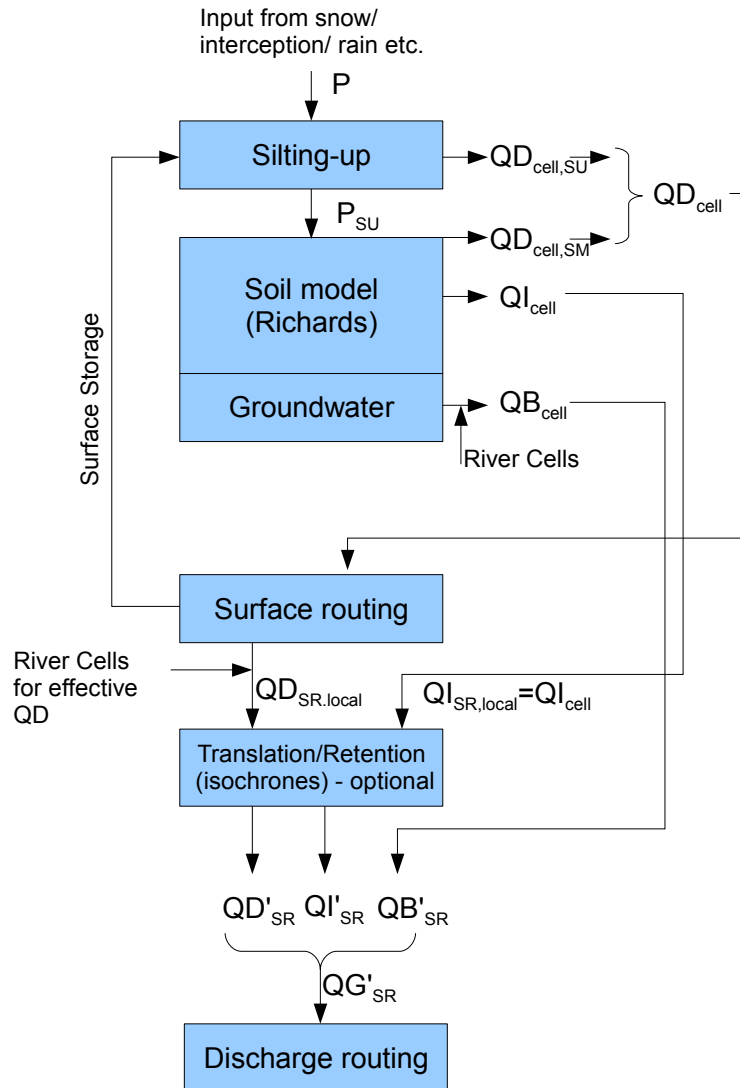


Figure 2: Flow chart for soil- and routing modelling including silting-up and surface routing (starting from version 8.1.01)

The other statistics files of the soil model will be written as usual, however: the valid input for balances are the statistical files of the surface routing model. Again: the direct runoff from soil model is NOT valid for balancing when surface routing is used, since it contains the surface storage of the silting up model. The surface routing model fits between the soil model and the (optional) translation/retention of the surface runoff and interflow using two single linear reservoir cascades. Surface runoff on a cell will only count as surface runoff for the statistics, if the cell is a valid river cell. All other cells will store the routed surface runoff for further routing until a river cell is reached. This implies the presence of a river network grid with at least on valid river cell per subbasin, otherwise no surface runoff will leave the subbasin at all. Interflow and baseflow are transferred untouched to the discharge routing module, so these two statistic files will be identical. The resulting total runoff  $QG'_{SR}$  is calculated as the sum of the routed surface runoff  $QD'_{SR}$  (optionally transformed by the single linear reservoir [SLR] cascade),  $QI'_{SR}$  (as well optionally

transformed by a SLR-cascade), and  $QB'_{SR}$ . As described in the overview, this reservoir cascade can be virtually disabled by setting all cells in the flow time grid to 1 (so only one flow time zone will be created) and setting the single linear reservoir recession constants for direct and/or interflow to a very small value. It is not possible to switch the reservoir cascade off for one flow component only. However, the direct runoff will enter the reservoir cascade only when it is routed on the surface and reaches a river cell. Thus, the effect of the reservoir cascade is rather small if there are only a few river cells at the outlet of the sub-catchments.

Between the two extremes of using no surface runoff and no silting up on one side and using both modules on the other side, there are two more possibilities: both modules could be used without the other module, so a schema as shown in Figures 3 and 4 is perfectly possible.

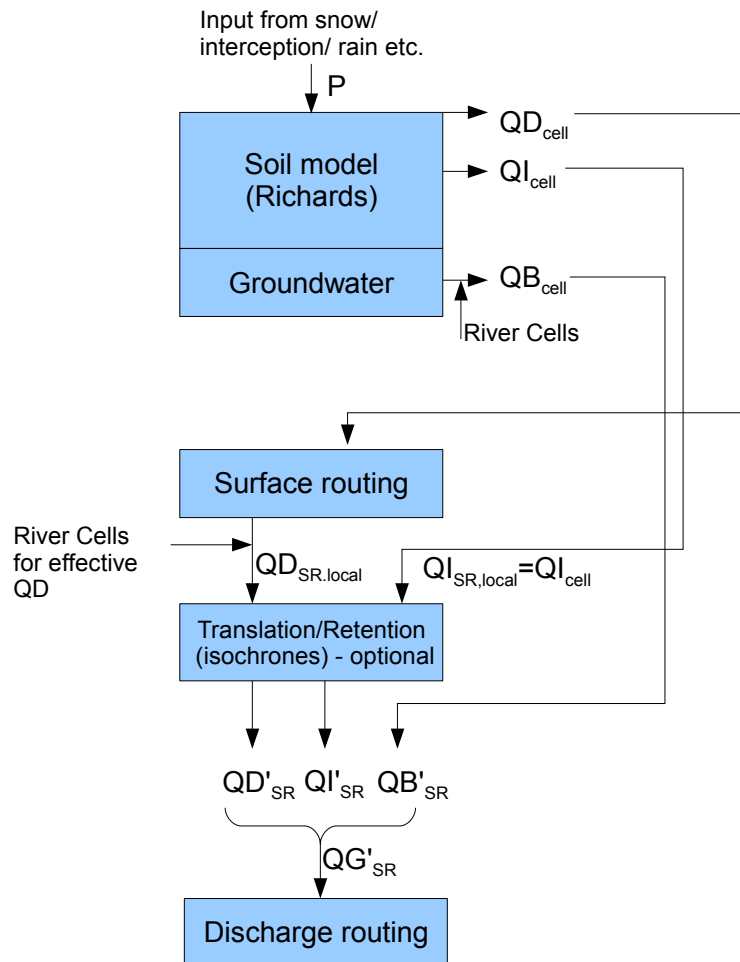


Figure 3: Flow chart for soil- and routing modelling including surface routing but without silting-up

If no silting-up model is used, then the input into the soil model is the traditional input from the interception storage or snow model or precipitation. Since no other surface runoff component is calculated, the entire surface runoff is generated in the soil model (exception: surface runoff from snow may be there, but this is calculated in the soil model as well). All other flows are calculated as described before. The surface routing storage which is used in both models, surface routing and silting-up, is used in the surface routing model only.

Figure 4 delineates the application of the silting-up model *without* using the surface routing model. Again, the surface routing storage is used by one model component only (surface routing and silting-up module will check during initialisation if the surface routing storage already exists and

create it, if required).

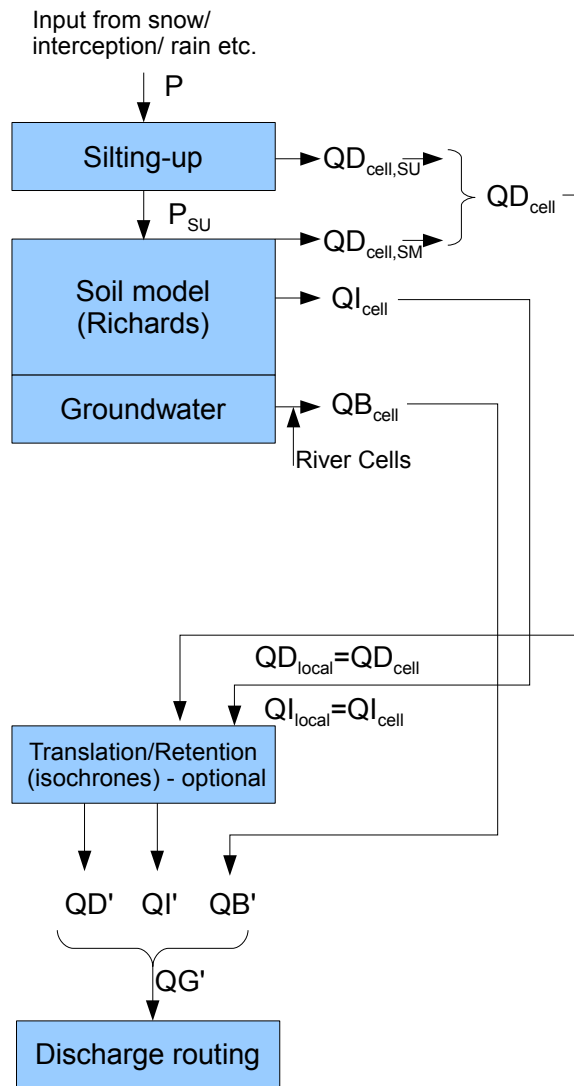


Figure 4: Flow chart for soil- and routing modelling including silting-up but without surface routing

All fluxes from the soil model are identical to the traditional model scheme, except for the two surface runoff components which are internally added to a total surface runoff.

## Theory of the Silting-up module

The core equation to estimate the potential infiltration rate is given in equation (1):

$$i_{pot}(t) = (i_0 - i_\infty) \cdot e^{-C_v \cdot E_{kin}(t)} + i_\infty \quad (1)$$

where

- $i_{pot}(t)$  potential infiltration rate at time  $t$  [mm/h]
- $i_0$  start infiltration rate (fix at 65 mm/h)
- $i_\infty$  end infiltration rate (see below) [mm/h]
- $C_v$  silting-up disposition depending on grain size distribution (see below) [m<sup>2</sup>/J]
- $E_{kin}(t)$  measure for cumulated kinetic energy at time  $t$  (see below) [J/m<sup>2</sup>]

The total potential infiltration for a model time step is calculated after:

$$I_{pot}(t) = i_{pot}(t) \cdot \Delta t \quad (2)$$

where

- $I_{pot}(t)$  potential infiltration [mm]
- $\Delta t$  model time step [h]

Applied to the actual input into the cell consisting of precipitation and already present water from inflow or local storage, the actual infiltration can be calculated as:

$$\begin{aligned} I_{act}(t) &= p(t) + q_{in}(t) & \text{if } I_{pot}(t) > (p(t) + q_{in}(t)) \\ I_{act}(t) &= I_{pot}(t) & \text{if } I_{pot}(t) \leq (p(t) + q_{in}(t)) \end{aligned} \quad (3)$$

where

- $I_{act}(t)$  actual infiltration for time step  $t$  [mm]
- $p(t)$  precipitation (input) into the cell for time step  $t$  [mm]
- $q_{in}(t)$  inflow from other cells and local storage from last time step [mm]

The surface runoff for the actual time step  $QD_{SU}(t)$  results to:

$$QD_{SU}(t) = p(t) + q_{in}(t) - I_{act}(t) \quad (4)$$

To this infiltration excess additional surface runoff may be added by the Richards approach in the unsaturated zone model. Thus, the actual infiltration is in fact the minimum of the infiltration rates calculated by Richards approach and by equations (1) to (3).

There are some entities from equation (1) which must be estimated or parameterized separately.

The end infiltration rate  $i_\infty$  is calculated by:

$$i_\infty = 12.2 \cdot d_g^{0.52} \cdot f_d^{-0.64} \quad (5)$$

where

- $d_g$  geometric mean of grain size distribution [ $\mu\text{m}$ ] (see below)
- $f_d$  fraction of sand [weigh-%]

$$d_g = 10 \frac{c_y \cdot f_y + c_i \cdot f_i + c_d \cdot f_d + c_{s1} \cdot f_{s1} + c_{s2} \cdot f_{s2} + c_{s3} \cdot f_{s3} + c_{s4} \cdot f_{s4}}{200} \quad (6)$$

where

- $f_y$  fraction of clay [weight-%]
- $f_t$  fraction of silt [weight-%]
- $f_d$  fraction of sand [weight-%]
- $f_{s1}$  fraction of stones 2 - 6.3 mm [weight-%]
- $f_{s2}$  fraction of stones 6.3 - 20 mm [weight-%]
- $f_{s3}$  fraction of stones 20 - 63 mm [weight-%]
- $f_{s4}$  fraction of stones 63 - 200 mm [weight-%]
- all values of  $f_y..f_{s4}$  are given as relative values [-] in the control file
- $c_y..c_{s4}$  weighting factors for the respective  $f_y..f_{s4}$  defined as follows:

$$c_y = \log_{10}(0.04) + \log_{10}(2) = -1.0969 \quad (7)$$

$$c_t = 0.3326(\log_{10}2 + \log_{10}6.3) + 0.3348(\log_{10}6.3 + \log_{10}20) + 0.1704(\log_{10}20 + \log_{10}36) + 0.1622(\log_{10}36 + \log_{10}63) = 2.1004 \quad (8)$$

$$c_d = 0.1336(\log_{10}63 + \log_{10}100) + 0.2005(\log_{10}100 + \log_{10}200) + 0.3318(\log_{10}200 + \log_{10}630) + 0.3341(\log_{10}630 + \log_{10}2000) = 5.1004 \quad (9)$$

$$c_{s1} = (\log_{10}2000 + \log_{10}6300) = 7.1004 \quad (10)$$

$$c_{s2} = (\log_{10}6300 + \log_{10}20000) = 8.1004 \quad (11)$$

$$c_{s3} = (\log_{10}20000 + \log_{10}63000) = 9.1004 \quad (12)$$

$$c_{s4} = (\log_{10}63000 + \log_{10}200000) = 10.1004 \quad (13)$$

The silting-up disposition  $C_v$  is calculated by:

$$C_v = 0.013 \cdot f_d^{-1.03} \cdot d_g^{0.7} \cdot t_{cult}^{-0.19} \quad (14)$$

where

- $f_d$  fraction of sand [%] as above
- $f_y$  fraction of clay [%] as above
- $t_{cult}$  time since last soil cultivation ( $\min(t_{cult})=0.3$ ) [d]

If silting up is calculated on a day with soil cultivation, the value of  $t_{cult}$  is set to 0.3 (not 0). If no tillage days are specified in the control file, the (only) tillage day is set to -1000, so  $t_{cult}$  will be 1000 for January, 1<sup>st</sup> and 1365 for December, 31<sup>st</sup>.

The cumulative kinetic energy  $E_{kin}(t)$  is calculated after the following schema:

$$\begin{aligned} e_n(t) &= 0 & \text{if } n(t) < 0.05 \\ e_n(t) &= 28.33 & \text{if } n(t) > 76.2 \\ e_n(t) &= 11.89 + 8.73 \cdot \log_{10}(n(t)) & \text{if } 0.05 \leq n(t) \leq 76.2 \end{aligned} \quad (15)$$

where

- $e_n(t)$  energy of precipitation for time step t [J/(m<sup>2</sup>·mm)]
- $n(t)$  precipitation intensity [mm/h]

$$\begin{aligned} E_{kin}(t) &= e_n(t) \cdot n(t) \cdot (1 - f_{res}(t)) \cdot \Delta t & \text{if } t_{RP} > t_{RP,min} \\ E_{kin}(t) &= E_{kin}(t - \Delta t) + e_n(t) \cdot n(t) \cdot (1 - f_{res}(t)) \cdot \Delta t & \text{if } t_{RP} \leq t_{RP,min} \end{aligned} \quad (16)$$

where

- $E_{kin}(t)$  cumulated kinetic energy of precipitation at time t [J/m<sup>2</sup>]
- $f_{res}$  fraction of soil covered by residues, plants and stones (total) [1/1]
- $\Delta t$  duration of a time step [h]
- $t_{RP}$  rain pause (time since last precipitation with no inflows and local storage = 0) [d]
- $t_{RP,min}$  minimum rain pause (value given in control file for each subbasin) [d]



## Control file extensions for Silting-up module

Starting with WaSiM 8.01.01, a new submodel parameter set named [SiltingUpModel] is added to the control file, which must be present in any control file (if silting-up is not used: at least the name of the submodel and the switch-parameter set to 0 must be present).

The example below shows a possible parameterization. The grid containing the surface water storage is shared between the two submodels “silting-up” and “surface routing”. It is created internally. To be able to write this grid to disk (which may be a good idea, especially when coupling different model runs, since the surface storage must be read in at the start of the coupled model run), the surface storage grid should be defined as variable grid with a default value of 0 (and a replace-nodata-value of 1).

The meaning of the parameters is described in the following example. Minimum rain pause must be defined for each subbasin. However, the abbreviation as commonly used in [unsatzonmodel] can be used here, too: omitted subbasin codes and parameter values are filled in automatically by the control file reader. So, a unique value for all subbasins in the example could be defined by simply set only the value for subbasin 1 (subbasin 1 must exist, however).

```
$set $SurfStorSiltingUp    = sftsu//$grid//.$suffix
$set $SDISPgrid           = sdis//$grid//.$suffix
$set $RPAUSgrid          = paus//$grid//.$suffix
$set $EKIN_grid           = ekin//$grid//.$suffix
$set $TSBB_grid           = tsbb//$grid//.$suffix
$set $QDSU_grid           = qdsu//$grid//.$suffix

→ optional entry in variable grids: if the grid is not listed here, it is created internally and
cannot be written
[variable_grids]
...
$outpath//$SurfStorSiltingUp SurfStorSiltingUp 1 0 # storage for surface runoff
$Writegrid                  # Writegrid for this grid
$readgrids                  # 0 or 1
...

[SiltingUpModel]
1                            # 0=ignore this module, 1 = run the module
$time                       # duration of a time step in minutes
$outpath//sdis//$grid//.$code//$year $hour_mean # statistics for silting up disposition
$outpath//qdsu//$grid//.$code//$year $hour_mean # direct discharge from silting up module
$outpath//$SDISPgrid        # grid with actual silting up disposition
$Writegrid                  # writegrid for this grid
$outpath//$RPAUSgrid        # grid with actual rain pause length
$Writegrid                  # writegrid for this grid
$outpath//$EKIN_grid        # grid with actual cumulated kinetic energy of the event
$Writegrid                  # writegrid for this grid
$outpath//$TSBB_grid        # grid with actual time since last soil tillage
$Writegrid                  # writegrid for this grid
$outpath//$QDSU_grid        # grid with direct runoff from silting up model
$Writegrid                  # writegrid for this grid
1 2 3 4 5 6 7 8 9 10 11 12 13 # range for subbasin codes
1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 # minimum Rainpause (in days)
$readgrids                  # readgrid code 0 do not read, 1 = read grids
```

Also, the land use table should contain the days of soil cultivation:

```
6 agriculture {method = VariableDayCount;
               SoilTillage = 90 240; # optional set of 1..n Julian days, depicting days
                                   with soil tillage. Important for silting up model
.....
```

If the model starts prior to the first tillage day, then the last tillage day is assumed to be the same JD in the last year, so in the above example, the last soil tillage has been on day 240 in the old year (which was not modelled), so the time since last tillage is  $365-240 = 125$  plus the actual Julian day (126 for day one).

The last required control file extension is a set of grain size distribution values which should be added to each soil type in the soil table.

The specific parameter is called GrainSizeDist and consists of 7 values:

- volumetric fraction of sand (default = 0.3333)
- volumetric fraction of silt (default = 0.3333)
- volumetric fraction of clay (default = 0.3334)
- volumetric fraction of stones 2-6.3 mm (default = 0)
- volumetric fraction of stones 6.3-20 mm (default = 0)
- volumetric fraction of stones 20-63 mm (default = 0)
- volumetric fraction of stones 63-200 mm (default = 0)

```
1 sand_(S) {method = MultipleHorizons;
  FCap = 6.21; mSB = 38.5; ksat_topmodel = 8.25E-5; suction = 385; # optional for Topmodel
  GrainSizeDist = 0.75 0.1 0.05 0.05 0.03 0.01 0.01; # optional: when using silting up
model, #the grain size fractions for sand, silt, clay, and Stones1..4 must be given here.
#Stones1 = 2-6.3mm, Stones2=6.3-20mm, Stones3=20-63mm, Stones4=63-200mm.
  PMacroThresh = 1000; # precipitation capacity thresholding macropore runoff in mm/hour
...
}
```

## Theory of the Surface routing module

Within the surface routing module we try to apply a kinematic wave approach for unsteady conditions including backwater effects. This would be basically impossible if we would not use a quasi-stationary approach. The basic concept is this:

- We divide the time step in as much sub steps as required to approximate unsteady conditions as good as possible
- One important parameter for an automated subdivision is the minimum flow time, the water needs from one cell to the next cell. The internal time step should not exceed this time since the inflow/outflow balancing only accounts for direct neighbors.
- When strong backwater conditions exists (very flat regions with shallow gradients but a lot of water flowing into such regions), the sub time step may be manually limited to force the model to work with smaller steps
- For each sub time step, the parameters like gradient and roughness and the inflow data are newly calculated. Then, the fluxes are routed.
- Using this schema, it is well possible that the gradient changes from one sub time step to the next, so the next sub time step will result in a different flow velocity and/or flow direction (or even in a reversed flow direction). To minimize numerical oscillations, the sub time step has to be sufficiently small compared to the normal time step. So it may be a good idea to set the maximum allowed sub time step to  $1/4^{\text{th}}$  to  $1/10^{\text{th}}$  of the normal time step.

The following explanations describe the scheme, WaSiM-ETH uses to implement this concept including the used equations (Source code encompasses appr. 2500 lines of C++-Code).

1. First of all, the input for each raster cell is taken from the preceding modules. This is usually the surface runoff calculated in the unsaturated zone model and/or in the silting-up model. Since the input is given as a total amount for the entire time step without differentiation between newly generated surface runoff (from infiltration excess or exfiltrated groundwater) and water which was already stored on the land surface from preceding time steps, it is not evenly distributed to all sub time steps but used as bulk input. This means, that there will be a tendency for higher water levels in the beginning of an event. But the flow velocities of runoff generated by from rain-runoff only are rather small compared to the velocities, stored water will allow (because of the much deeper water bodies). The disadvantage of the bulk input will be small compared to the numerical simplification.

The input is defined as  $Q_{in,total}$ , given in mm per time step.

2. Now, the flow velocities must be estimated. This is a rather complex task, since it incorporates roughness estimation, slope and aspect calculation (all dynamically!) and multiple flow paths definition (among others).

The slope or gradient will be determined for each cell as maximum positive difference between the elevations of the central cell and the eight neighboring cells (with a reduction factor of  $\sqrt{2} = 1.414$  for diagonal directions). Directions in the following equations are coded as:

North	=	1
North-East	=	2
East	=	3
South-East	=	4

South	=	5
South-West	=	6
West	=	7
North-West	=	8

The slope is calculated by:

$$S = \underset{i=1}{\overset{8}{\text{Max}}} \left( \frac{(h_0 + h_{w,0}) - (h_i + h_{w,i})}{c_s \cdot f_i} \right) \quad (17)$$

where	$S$	slope [-]
	$h_0$	absoulte elevation of the actual cell from the digital elevation model [m]
	$h_{w,0}$	water level ( $Q_{in,total}/1000$ ) for the actual central cell [m]
	$h_i$	absolute elevation from digital elevation model for the neighboring cell $i$ with $i$ as a code for the direction as given above
	$h_{w,i}$	water level ( $Q_{in,total}/1000$ ) for cell $i$ [m]
	$c_s$	cell size [m]
	$f_i$	direction dependent weighing factor: orthographic directions: 1.0 diagonal directions: $1.4142 = \sqrt{2}$

The values for  $h_{w,0}$  and  $h_{w,i}$  are only used when the method is set to 3 (multiple plow paths regarding the water level) or 4 (single flow path regarding water level). Methods 1 (multiple flow paths without regarding water level) and 1 (single flow paths without regarding water level) will set both values to 0.

The aspect or direction of exposition  $\Omega$  [° from north] is defined by the direction of the steepest slope.

Special cases:

**a) multiple slopes with the same value:**

Although this case will be rare since even the slightest difference in water levels will give a unique main flow direction for methods 3 and 4 , it may occur when using methods 1 or 2. The maximum slope  $S$  can still be calculated, but the aspect must be identified differently. Here, the approach used for the topographic analysis is used:

$$\Omega = 180 - \arctan(H/G) + 90(G/|G|) \quad (18)$$

where	$\Omega$	aspect angle, rounded to the nearest multiple of 45° (matching one of the directions 1...8 coded above)
	$G$	South-North-gradient $(h_5 - h_1)/2c_s$ [m]
	$H$	West-East-gradient $(h_7 - h_3)/2c_s$ [m]

**b) no slope at all:**

$S$  will be set to 0,  $\Omega$  will be set to 0 (north, direction code 1)

**c) missing cells at the border:** the actual grid cell is at the border of the model domain (nodata-values or no values at all are partly surrounding the actual cell):

Equation (17) is applied for valid surrounding cells only . If special cases a) or b) happen,

they are solved as described above. Missing elevations for equation (18) are taken from the central cell, if required, with an additional difference regarding the gradient from the central cell to the cell in the opposite direction (e.g., when no cell to the north is present but to the south, the elevation of the northern cell is estimated by taking the elevation of the central cell and adding the elevation difference between the central and the southern cell; same for western or eastern cell).

When knowing the main flow direction (aspect angle in 45° steps) and its gradient, the multiple flow paths can be identified. WaSiM detects multiple flow path based on two assumptions:

- a) the plan curvature of the actual cell is negative → the flow is diverging
- b) multiple flow paths are possible to the left and to the right of the main flow direction only (in 45°-steps). Exception: if gradients to all directions are identical, then flow into all directions is possible.

The plan curvature  $\omega$  is calculated by:

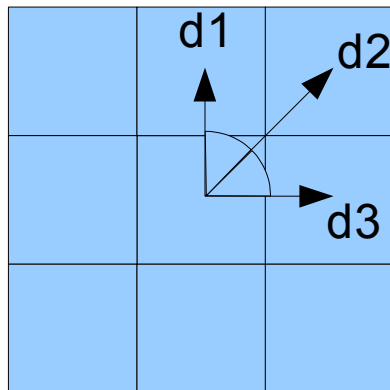
$$\omega = 2 \frac{DH^2 + EG^2 - FGH}{G^2 + H^2} \quad (19)$$

where

$D$	$((h_1 + h_5)/2 - h_0)/c_s^2$ see equation (17)
$E$	$((h_3 + h_7)/2 - h_0)/c_s^2$ see equation (17)
$F$	$(h_2 + h_6 - h_4 - h_8)/4c_s^2$ see equation (17)
$G$	$(h_5 - h_1)/2c_s$ see equation (17)
$H$	$(h_7 - h_3)/2c_s$ see equation (17)

If the plan curvature is positive, the main flow direction will be the only flow path for that cell. For negative plan curvatures, the flow paths 45° to the left and 45° to the right are equally weighted. They are checked for a positive gradient. If the gradient of one or both minor flow directions is negative or 0, then the respective flow direction weighting is set to 0. The following drawing shows the main principles: d2 is the main flow direction with weight  $w_2=1.0$ . d1 and d3 are minor flow directions with weights  $w_1=w_3=0.7071$ , according the  $\cos(45^\circ)$ . All weights are normalized to  $w_1 + w_2 + w_3 = 1.0$ , so the resulting weights are

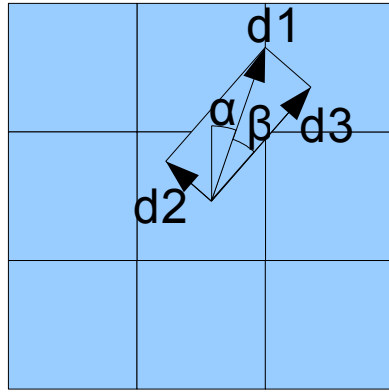
$$\begin{aligned} w_1 &= 0.2929 \\ w_2 &= 0.4142 \\ w_3 &= 0.2929 \end{aligned} \quad (20a)$$



For aspects not fitting into the 45°-raster (i.e. Calculated by eq. (18)), the weights  $w_1$  to  $w_3$  are calculated by:

$$\begin{aligned}
 w_1 &= \cos(\alpha) \cdot \cos(\beta) & \text{if } \alpha \geq 0 \\
 w_1 &= \cos(\alpha) \cdot \sin(\beta) & \text{if } \alpha < 0 \\
 w_2 &= \cos(\alpha) \\
 w_3 &= \cos(\alpha) \cdot \sin(\beta) & \text{if } \alpha \geq 0 \\
 w_3 &= \cos(\alpha) \cdot \cos(\beta) & \text{if } \alpha < 0
 \end{aligned}
 \tag{20b}$$

where  $\alpha$  angle between the main flow direction (d1 in the drawing below) and the nearest flow direction class (1 to 8, as given above), range between  $-22.5^\circ \dots 22.5^\circ$   
 $\beta$  complementary angle of  $\alpha$  to  $\pm 45^\circ$  (because the next minor flow direction d2 or d3 – depending on the sign of  $\alpha$  - will always have an angle of  $45^\circ$  against the main flow direction, see drawing below)



As for 45°-steps-aligned aspects, weights are normalized to 1.

After identification of the flow directions and their weightings, the flow velocities for each direction are calculated. The basic equation is the Manning-Strickler equation:

$$v_i = M \cdot R_H^{2/3} \cdot S_i^{1/2} \tag{21}$$

where  $v_i$  flow velocity in direction  $i$  (as coded above)  
 $M$  roughness coefficient after Manning-Strickler (see below)  
 $R_H$  hydraulic radius = water depth  
 $S_i$  gradient in direction  $i$

Solving equation (21) is an iterative approach, since  $R_H$ ,  $v$  and partly  $M$  depend on each other. So the starting value of  $v_i$  will be  $v_{i,0} = 0.1 \text{ m/s}$ .

The starting value of  $R_H$  is set to

$$R_H = \frac{q_i}{v_{i,0} \cdot c_s} \quad \text{with} \quad q_i = \frac{Q_{in, total} \cdot c_s^2 \cdot w_i}{1000 \cdot \Delta t} \tag{22}$$

where  $q_i$  flow in direction  $i$  [ $\text{m}^3/\text{s}$ ]

$v_{i,0}$	initial flow velocity in direction $i$ [m/s]
$w_i$	weighting of flow in direction $i$ [-] as defined above
$\Delta t$	time step [s]
$Q_{in,total}$	total input in mm/ $\Delta t$

Also the roughness coefficient  $M$  may depend on water level, see the explanations below. The first value of  $R_H$  is then used to calculate  $v_{i,1}$  which in turn is used for a second estimation of  $R_{H,1}$  which is then used for  $v_{i,2}$ -calculation and so on until one of the iterations break conditions is reached:

$$\left(|v_{i,n} - v_{i,n-1}| \leq \Delta v_{max}\right) \vee (n > n_{max}) \quad (23)$$

where	$n$	iteration counter
	$n_{max}$	maximum number of iteration steps (parameter from control file)
	$\Delta v_{max}$	maximum difference between two subsequent iteration steps (parameter from control file)

Having calculated the flow velocity and the hydraulic radius, two more checks are done. First, the possibility of **preferential flow paths** formation must be checked. This is done by calculating the shear stress and compare it against a given critical shear stress. If the actual shear stress is greater then the critical shear stress, the flow is artificially concentrated by scaling  $R_H$  by a factor given in the control file. This changed  $R_H$  requires a recalculation of the iterative equations (21) and (22). The shear stress  $\tau$  [Pa] is calculated by:

$$\tau = \rho \cdot g \cdot R_H \cdot S_i \cdot \left(\frac{M}{M_g}\right)^{3/2} \quad (24)$$

where	$\rho$	density of water at 20°C, here taken as 1000 [kg/m <sup>3</sup> ]
	$g$	gravitational acceleration = 9.815 m/s <sup>2</sup>
	$M$	roughness coefficient as calculated here for certain land uses (see below) in m <sup>1/3</sup> /s
	$M_g$	roughness coefficient for bare ground as given in the control file [m <sup>1/3</sup> /s]

The critical value  $\tau_{crit}$  can be given as cell-by-cell values as a standard grid or, if no such grid is available, is set to 0.9 per default. If  $\tau > \tau_{crit}$ ,  $R_H$  is scaled by the concentration factor from the control file and eq.s (21) and (22) are applied for the changed  $R_H$  again.

The second check deals with **too slow velocities** and thus too large hydraulic radiuses. If the gradient is very shallow or the flow velocity is very small because of low  $M$ -values, the iteration may result in  $R_H$ -values which are larger than the total input  $Q_{in,total}$  was. Since this is physically impossible, the hydraulic radius must be reduced to the physically maximum. Thus, the flow velocity will changed reduced, too, so equations (21) and (22) will be applied once more. In this third iteration, the  $R_H$ -value is kept constant, so the iteration is usually only a one-step iteration, except when  $M$  changes (see below).

The flow velocity is now used to calculate the **effective fluxes** in m<sup>3</sup>/s:

$$q_{i,eff} = \frac{v_i \cdot c_s \cdot R_H}{f_{conc}} \quad (25)$$

where  $q_{i,eff}$  effective flux in direction  $i$  [ $m^3/s$ ]  
 $f_{conc}$  concentration factor as parameter from the control file (or 1, if no channel generation takes place)

### Calculation of roughness coefficient $M$ :

There are three different algorithms for estimating  $M$ :

- **Roughness coefficient  $M$  for other land use than cultivated (e.g. Forest)**

$M$  is simply taken from the control file als  $M_g$  ( $g$ =ground).

- **Roughness coefficient  $M$  for conserving land use:**

It is assumed, that the crop parameters are neglectable for the total roughness. Conserving cultivation doesn't use ploughs and the residues remain on the field (or mulch will even be thrown out).  $M$  depends on the flow velocity  $v_i$  and on the hydraulic radius  $R_H$ . Starting point is the Reynolds number, a measure for the intrinsic friction due to kinematic viscosity of water:

$$Re = \frac{4 \cdot v_i \cdot R_H}{\nu} \quad (26)$$

where  $Re$  Reynolds number  
 $\nu$  kinematic viscosity at  $20^\circ C = 1.0 \cdot 10^{-6} m^2/s$   
 $R_H$  hydraulic radius [m],  $4 \cdot R_H =$  characteristic length for  $Re$

The roughness parameter  $M$  is then calculated by:

$$n_r = \frac{a \cdot f_{veg}^b}{Re^c} \rightarrow M = \frac{1}{n_r} \quad (27)$$

where  $n_r$  roughness coefficient [ $s/m^{1/3}$ ]  
 $a, b, c$  crop specific coefficients [-]  
 $f_{veg}$  fraction of covered soil [%]

As can be seen by equations (26) and (27), the ground roughness  $M_g$  has no effect on  $M$  here. Only the plant covered fraction (ideally: the fraction covered by residues) of the soil (taken from land use table) and the crop specific coefficients are parameters for  $M$ . The three empiric parameters  $a$ ,  $b$  and  $c$  are given in Table 1 (GILLEY):

Table 1: coefficients for calculating roughness coefficient  $M$  for conserving cultivation

Crop type	$Re < 20000$			$Re \geq 20000$		
	a	b	c	a	b	c
corn/maize	0.00496	0.89200	0.03110	0.00519	1.20000	0.17700
cotton	0.00896	0.67800	0.00930	0.00473	0.70000	-0.03260
peanuts	0.02730	0.70300	0.19100	0.00773	0.41100	-0.05090



Crop type	<i>Re</i> < 20000			<i>Re</i> ≥ 20000		
	a	b	c	a	b	c
pine(-needles)	0.00539	1.04000	0.19200	0.000332	1.23000	0.00311
sorghum	0.07850	0.48800	0.01980	0.03	0.71400	0.18900
soybeans	0.04510	0.91300	0.35800	0.00159	0.96100	0.05100
sunflower	0.04130	0.43900	0.08930	0.00411	0.85800	0.03300
wheat	0.00207	1.46000	0.30200	0.000192	1.03000	-0.09780
others	0.01890	0.71200	0.14200	0.01890	0.71200	0.14200

The coefficients given in table 1 must be set in the control file per land use type. Default values (if omitted) are the parameters for crop type „others“.

- **Roughness Coefficient *M* for flow through vegetation (orig.: non-conserving land use)**

Non-conserving cultivation is conventional cultivation using ploughs. However: This approach should mainly be used for flowed through vegetation (bushes, woods etc.). *M* is assumed to be a result of the grounds roughness and the roughness of vegetation, the water flows through. The approach is based on the algorithms suggested by PASCHE, an improvement of the approach by LINDNER. Vegetation roughness is effectively used for flow depths above a specified value (in the control file). This is because of instabilities of the algorithms for very shallow water depths.

Ground roughness is calculated by

$$\lambda_g = \frac{8 \cdot g}{R_H^{1/3} \cdot M_g^2} \quad (28)$$

where  $\lambda_g$  roughness of the bare ground [-]  
 $M_g$  Manning-Strickler coefficient for bare ground, given in the control file [m<sup>1/3</sup>/s]  
 $R_H$  hydraulic radius [m]  
 $g$  gravitational constant = 9.81 m/s<sup>2</sup>

Vegetation roughness is calculated by the following schema.

At first, a few parameters will be estimated:

$$Re_{veg} = \frac{v \cdot d_p}{\nu} = \frac{v \cdot d_p}{10^{-6}} \quad (29)$$

where  $Re_{veg}$  Reynolds number for flowed through vegetation [-]  
 $v$  flow velocity (iterative) [m/s]  
 $d_p$  stem diameter of the flowed through vegetation  
 $\nu$  kinematic viscosity of water at 20°C (=10<sup>-6</sup>) [m<sup>2</sup>/s]

The form drag coefficient of a single circular cylinder is derived from  $Re_{veg}$ :

$$\begin{aligned} c_{w\infty} &= 3.07 \cdot Re_{veg}^{-0.168} & \text{if } Re_{veg} < 800 \\ c_{w\infty} &= 1.0 & \text{if } 800 \leq Re_{veg} < 8000 \\ c_{w\infty} &= 1.2 & \text{if } 8000 \leq Re_{veg} < 10000 \end{aligned} \quad (30)$$

where  $c_{w,\infty}$  form drag coefficient of a single circular cylinder [m]

For  $Re_{veg} > 10000$ , the value of  $c_w$  from the control file is used instead (form drag coefficient for a circular cylinder in a group), default value is around 1.5

Two more intermediate entities are now estimated: the wake length  $a_{WL}$  and the wake width  $a_{WW}$  characterizing the dimensions of the wake behind a circular cylinder:

$$a_{WL} = 128.9 \cdot c_{w\infty} \cdot d_p \cdot \left( 1 + \frac{g \cdot a_{WL} \cdot I_E}{v^2 / 2} \right)^{-2.14} \quad (31)$$

where  $a_{WL}$  wake length [m]  
 $I_E$  gradient (usually energy curve gradient, but here set to ground slope)  
 $v$  flow velocity

Since  $a_{WL}$  is dependent on itself, equation (31) can only be solved iteratively. Starting parameters are  $a_{WL,1} = 50 \cdot d_p$ , and  $\Delta a_{WL} = 1.0$ . The control file contains two parameters for controlling the iteration: maximum wake length iteration difference  $\Delta a_{WL,max}$  (e.g. 0.001m) and the maximum number of iterations, e.g. 40.

As some analyses have shown, the iteration may not always converge. However, WaSiM needs to have stable results. To ensure this, the effective gradient is limited:

$$\frac{c_{w\infty} \cdot I_E}{v^2} > 0.04 ? \rightarrow I_E = \frac{v^2}{25 \cdot c_{w\infty}} \quad (32)$$

Now,  $a_{WW}$  can be identified:

$$a_{WW} = 0.24 \cdot a_{WL}^{0.59} \cdot (c_{w\infty} \cdot d_p)^{0.41} \quad (33)$$

where  $a_{WW}$  wake width [m]

To calculate the form drag coefficient of the vegetation  $c_{WR}$  (not to be taken for  $c_{w,\infty}$ , the form drag coefficient for a single uninfluenced circular cylinder!), the relative flow velocity as relation between the velocities before and after a cylinder is estimated, using  $a_{WL}$  and  $a_{WW}$ :

$$\left( \frac{v_b}{v_a} \right)^2 = 1.15 \cdot \left( \frac{a_{WL}}{a_x} \right)^{-0.48} + 0.5 \cdot \left( \frac{a_{WW}}{a_y} \right)^{1.1} \quad (34)$$

where  $v_b$  flow velocity before the obstacle [m/s]

$v_a$	flow velocity after the obstacle [m/s]
$a_x$	interspace between crops in flow direction, from control file [m]
$a_y$	interspace between crops perpendicular to flow direction, from control file [m]

The last required parameter for calculating  $c_{WR}$  is  $\Delta c_W$ , a correction term accounting for scaling effects which depends on the Froude number  $Fr$ :

$$Fr = \frac{v}{\sqrt{(g \cdot R_H)}} \quad (35)$$

where  $Fr$  Froude number:  $Fr < 1$ : laminar flow,  $Fr > 1$ : turbulent flow [-]

$$\Delta c_W = \frac{2}{Fr^2} (1-y) \quad \text{with} \quad y = 2 \cdot Fr \cdot \frac{\left(y - \frac{a_y}{a_y - d_p}\right)}{y^2 - 1} \quad (36)$$

where  $y$  internal intermediate result

The identification of  $y$  is again an iterative process (using 0.0001 as maximum iteration error and taking the maximum number of iterations from the same parameter as the  $a_{WL}$ -iteration). To ensure the iterations convergence, some checks and adjustments are required:

$$a_y < 5 \cdot d_p ? \rightarrow a_y := 5 \cdot d_p \quad (37)$$

$$Fr \cdot \left(\frac{a_y}{a_y - d_p}\right)^3 \geq \frac{2}{3} ? \rightarrow Fr = \frac{2}{3 \cdot \left(\frac{a_y}{a_y - d_p}\right)^3} \quad (38)$$

$$Fr < 0.05 ? \rightarrow Fr = 0.05 \quad (39)$$

The value of  $\Delta c_W$  is then calculated after equation (36) (first part). Now  $c_{WR}$  can be calculated by:

$$c_{WR} = 1.31 \cdot c_{W\infty} \cdot \left(\frac{v_b}{v_a}\right)^2 + \Delta c_W \quad (40)$$

$c_{WR}$  is now used to determine the vegetation roughness finally:

$$\lambda_{veg} = \frac{4 \cdot R_H \cdot d_p}{a_x \cdot a_y} + \Delta c_W \quad (41)$$

The **total roughness** is the sum of **ground roughness** and **vegetation roughness**:

$$\lambda_{total} = \lambda_{veg} + \lambda_g \quad (42)$$

which leads finally to the Manning roughness coefficient  $M$  as:

$$M = \sqrt{\frac{8 \cdot g}{\lambda_{total} \cdot R_H^{1/3}}} \quad (43)$$

3. **Time step splitting:** Having calculated the flow velocities and the fluxes in m<sup>3</sup>/s, the next step will be the calculation of the dynamic time step. Since all fluxes should flow to the neighboring cell only, the sub time step must be as short as required to allow the fastest flux to flow to the next cells border only.

$$t_{step} = \min_{i=1}^n \left( \frac{c_s \cdot c_{corr}}{v_i} \right) \quad \text{with} \quad \begin{array}{l} c_{corr} = 1.0 \text{ for flow direction 1, 3, 5 and 7} \\ c_{corr} = \sqrt{2} \text{ for flow direction 2, 4, 6 and 8} \end{array} \quad (26)$$

where  $t_{step}$  sub time step in seconds  
 $n$  number of active cells in the grids  
 $c_s$  cell size [m]  
 $c_{corr}$  correction factor accounting for orthogonal or diagonal directions  
 $v_i$  flow velocity in cell  $i$

WaSiM provides the possibility to limit both, the upper and the lower limit of the sub time step  $t_{step}$ . The duration of a time step is the absolute maximum for  $t_{step}$ , but the control file contains two parameters for the allowed range of the sub time step, one for the minimum sub time step ( $t_{step,min}$ ) and one for the longest allowed sub time step ( $t_{step,max}$ ). When setting  $t_{step,min}$  to a significant larger value than the shortest flow time would suggest, the result will NOT be a complete unpredictable runoff concentration but rather a more retarded one – the fastest fluxes will only reach the next cell with a longer sub time step (and not the cell after next or even further). The limitation of  $t_{step,max}$ , however, to some smaller value than the fastest flow time suggests, may have positive effects: Since WaSiM uses a kinematic wave approach stating stationary conditions, backwater effects could only be accounted for when using smaller sub time steps. Thus, a very small gradient will not lead to oscillations in the flow directions.

4. The next step – the balancing of the fluxes – is optimized for parallel computing. Since WaSiM uses multiple flow paths and a cell may receive water from multiple other cells as well as it may send water to multiple other cells, the balancing is separated into two subsequent steps, each of which is parallelized. Otherwise, the adding of water from outflow into another cell could break the thread safeness (when two threads would add water into the same third cell, the addition of at least one flux would use the wrong initial value (i.e. the same initial value than the other thread is just using) and it would overwrite the result of the other thread, so water would be virtually lost.

1. Calculating the totals of inflowing and outflowing water masses and tracer masses

$$q_{in,total} = \sum_{i=1}^8 \hat{q}_{out, (i+3) \bmod 8 + 1} \quad (27)$$

$$q_{out,total} = \sum_{i=1}^8 q_{out,i} \quad (28)$$

where  $q_{in,total}$  total inflow into the actual cell from neighboring cells [m<sup>3</sup>/s]

- $\hat{q}_{out, (i+3) \bmod 8 + 1}$  outflow of the cell laying in direction  $i$  into the actual cell – the flow direction for that cell will be calculated by  $(i+3) \bmod 8 + 1$ , so e.g. the upper right cell with index 2 for the actual cell will contribute with its outflow in direction  $((2+3) \bmod 8)+1 = 6$ , the upper left cell with index 8 with regard to the actual cell will contribute with its outflow in direction  $((8+3) \bmod 8)+1 = 4$  a.s.o.
- $q_{out, total}$  total outflow from the actual cell
- $q_{out, i}$  outflow from the actual into neighboring cell in direction  $i$

2. Balancing the actual water storage content for each cell:

$$m_{ss, t + t_{step}} = m_{ss, t} + \frac{q_{in, total} + q_{out, total}}{c_s^2} \cdot \Delta t \quad (29)$$

- where  $m_{ss}$  amount of water in the surface routing storage [mm or kg/m<sup>2</sup>] at time  $t$
- $\Delta t$  time step [s]

After balancing the water fluxes, the tracer masses are balanced using the actual fluxes and the known tracer concentrations in the surface runoff storage.

5. For cells marked with a valid value in the river depth- and river-width grids, the value of  $m_{ss}$  (which is usually equivalent to  $q_{in, total}$ ) will be added to the direct runoff grid ( $QD$ ). For all other cells, the storage content in  $m_{ss}$  is then taken as  $Q_{in, total}$  for the next sub time step
6. The sub time step counter is incremented and the schema described in equations (17) to (29) is processed again (except calculation of the sub time step duration in eq. (26)) until the entire time step is finished.

To optimize the performance, the flow velocity calculation is skipped when all newly calculated weights (in any of the 8 possible directions) did not change by more than 1%. Then, the old velocities are used, again. However, the first flow velocity calculation in a time step will be carried out in any case, only subsequent *sub time steps* of the same time step may be optimized this way.

7. At the end of a time step, the total amounts of surface runoff and interflow are passed with its cell values into the single linear storage cascade to consider the translation and retention effect of the river network. To switch off this feature, simply no flow time grid has to be provided in the control file as standard grid.
8. The outputs from the single linear storage cascades for surface runoff ( $QD$ ), interflow ( $QI$ ) and base flow ( $QB$ ) (with  $QI$  and  $QB$  taken directly from the unsaturated zone model) are then superposed to the total runoff ( $QGES$ ) for each subbasin. This is the input for the routing model.
9. The content of the surface routing storage which did not run off into a river cell, will be taken as input to the silting-up module or, if no silting-up module is used, to the unsaturated zone module in the next time step.

## Control file extensions for Surface routing

Starting with WaSiM 8.01.01, a new submodel parameter set named [SurfaceRoutingModel] is added to the control file, which must be present in any control file (if surface routing is not used: at least the name of the submodel and the switch-parameter set to 0 must be present).

The example below shows a possible parameterization. The grid containing the surface water storage is shared between the two submodels “silting-up” and “surface routing”. It is created internally. To be able to write this grid to disk (which may be a good idea, especially when coupling different model runs, since the surface storage must be read in at the start of the coupled model run), the surface storage grid should be defined as variable grid with a default value of 0 (and a replace-nodata-value of 1).

The meaning of the parameters is described in the following example. There are two stacks, one for surface flow velocities and one for the amount of water flowing into each direction. These stacks are organized by flow direction: Level 1 contains the fluxes and velocities, resp., which are valid for direction 1 (North). Level 2 contains the fluxes and velocities valid for flow direction 2 (North-East) a.s.o. The 0-level contains the flux and velocity (in the respective stacks) in the main direction only.

The actual water level of all cells above sea level (value of DHM + surface storage) is stored in the 9<sup>th</sup> layer of the surface velocity stack.

```
[SurfaceRoutingModel]
1 # 0=ignore this module, 1 = run the module
$time # duration of a time step in minutes
3 # method: 1=MultipleFlowPaths for diverging areas, 2=single
(no line feed here in reality) (nearest direction as given by aspect), 3 = Multiple Flow Paths
(no line feed here in reality) regarding water level) 4 = single flow paths (regarding water level)
$outpath//qdsr//$grid//.$code//$year $hour_mean # direct discharge from surface routing module
$outpath//qisr//$grid//.$code//$year $hour_mean # interflow from surface routing module
$outpath//qbsr//$grid//.$code//$year $hour_mean # baseflow from surface routing module
$outpath//qgsr//$grid//.$code//$year $hour_mean # total discharge from surface routing module
$outpath//$surfspeed_grid # stack with actual flow velocity of surface flow in m/s, the 0-element
(no line feed here in reality) contains the speed in main direction (or single direction flow)
$Writegrid # writegrid for this grid
$outpath//$surfflux_grid # stack with actual flow amounts of surface flow in m^3/s, the 0-
(no line feed here in reality) element contains the flow in main direction
$Writegrid # writegrid for this grid
0.001 # maximum wake lenght iteration difference (if Delta_A_n1 < this
(no line feed here in reality) value, iteration for a_NL stops)
40 # maximum number of iterations for a_NL
0.0001 # maximum flow velocity iteration difference (if Delta v is less than
(no line feed here in reality) this value, iteration stops)
40 # maximum number of iterations for v
30 # shortest sub-time step in seconds
300 # longest allowed sub time step (even if flow travel times are longer,
(no line feed here in reality) the time step is subdivided into sub timesteps of this lenght) be
(no line feed here in reality) careful: tracers are mixed much faster when multiple sub time steps
(no line feed here in reality) are applied
0.04 # minimum water depth for regarding roughness of crops in m (shallower
(no line feed here in reality) sheet flow: only roughness of bare soil will be regarded)
3.0 # ConcentrationFactor takes into account the micro scale concentration
(no line feed here in reality) of flow pathes, flow will take place on a fraction of the cell only,
(no line feed here in reality) so the amount flowing per meter width will be multiplied by this
(no line feed here in reality) factor (1..n)
$readgrids # readgrid code 0 do not read, 1 = read grids
```

Furthermore, some land use specific parameters are used in the land use table. Here is an example with parameter description. The additional parameters for surface routing

are printed in blue/bold font:

```

6 agriculture      {method      = VariableDayCount;
  SoilTillage      = 90 240;# optional set of 1..n Julian days, depicting days with soil
                    tillage. Important for silting up model
  RootDistr        = 1.0; # parameter for root density distribution
  TReduWet         = 0.95; # relative Theta value for beginning water stress (under wet
                    conditions -> set >= 1 for crop which doesn't depend on an aeral
                    zone
  LimitReduWet     = 0.5; # minimum relative reduction factor of real transpiration when water
                    content reaches saturation. The reduction factor value will go down
                    linearly starting at 1.0 when relative Theta equals TReduWet (e.g.
                    0.95) to LimitReduWet when the soil is saturated (Theta rel = 1.0)
  HReduDry         = 3.45; # hydraulic head (suction) for beginning dryness stress (for water
                    content resulting in higher suctions, ETR will be reduced down to 0
                    at suction=150m)
  IntercepCap      = 0.3; # optional: specific thickness of the water layer on the leafes in mm.
                    if omitted here, the dedfault parameter from interception_model is
                    used
  StartVegetationPeriodForBalance = 3 ; # the sampling point in the following JD-Table when the
                    vegetation period starts, default = 0 (start of model
                    run)
  StopVegetationPeriodForBalance = 10 ; # the sampling point in the following JD-Table when the
                    vegetation period ends, default = n+1 (end of model run)
  JDVegetationResetForBalance = 1 ; # Julian day, when vegetetaion start and vegetation stop grids
                    are re-initialized to -1 (northern hemisphere: usually day 1)
  JulDays          = 15   46   74   105  135  166  196  227  258  288  319  349 ;
  Albedo           = 0.25 0.25 0.25 0.25 0.25 0.25 0.25 0.25 0.25 0.25 0.25 0.25;
  rsc              = 80   80   75   75   65   65   65   65   65   75   90   90;
  rs_interception  = 80   80   75   75   65   65   65   65   65   75   90   90;
  rs_evaporation   = 240  240  240  240  240  240  240  240  240  240  240  240;
  LAI              = 1    1    2    3    4    5    5    4    3    2    1    1;
  Z0               = 0.05 0.05 0.1  0.25 0.5  0.5  0.5  0.4  0.3  0.15 0.05 0.05;
  VCF              = 0.3  0.3  0.3  0.5  0.8  0.8  0.8  0.7  0.6  0.3  0.3  0.3;
  RootDepth        = 0.05 0.05 0.1  0.2  0.3  0.4  0.4  0.4  0.3  0.2  0.05 0.05;
  AltDep           = 0.025 0.025 0.025 0.025 0.025 0.025 -0.025 -0.025 -0.025 -0.025 -0.025;
# now some parameters for surface routing will follow. They are optional, the default value is
given at the start of the comment.
  SR_cultivation = nonconserving; # default: other; type of cultivating the surface; possible
                    values: conserving (c), nonconserving (nc), other (o), abbrev.
                    used in the following rows are for convenience only
(nc) SR_StemDiameter = 0.01 0.02 0.03 0.04 0.05 0.05 0.08 0.08 0.05 0.03 0.02 0.02;
    # default: 0.01 surface routing: diameter of the stalks or effective diameter of
    crop/tree stems (perpendicular to the streamlines); used for nonconserving land use
    (using ploughs)
(c) SR_ResidCovrg = 0.3 0.4 0.6 0.7 0.8 0.9 0.9 0.9 0.6 0.5 0.4 0.3;
    # default: 0.75 fraction of soil covered by vegetation, stones, and
    residuals (mulch); used for conserving land use only
(nc) SR_CropDistX = 0.3; # default: 0.3 distance of crops in flow direction; used for
    nonconserving land use (using ploughs) only
(nc) SR_CropDistY = 0.4; # default: 0.3 distance of crops perpendicular to flow direction;
    used for nonconserving land use (using ploughs) only
(nc/o) SR_K_ST = 10; # default: 10 Manning-Strickler-roughness for the soil surface in
    (m1/3)/s; used for nonconserving and other landuse only
(nc) SR_C_WR = 1.5; #default: 1.5 form drag coefficient for a circular cylinder in a
    group; used for nonconserving landuse only
(c) SR_rough_coeff1 = 0.00496 0.892 0.0311; # default: 0.0198 0.712 0.142; coefficients for
    k_st-calculation for conserving landuse, Reynolds
    number < 20000
(c) SR_rough_coeff2 = 0.00519 1.200 0.177 ; # default: 0.0198 0.712 0.142; coefficients for
    k_st-calculation for conserving landuse, Reynolds
    number >= 20000
}

```