# Sediment Transport Model

Regarding Garcia *et al.* (2015), the transport of fine sediment and deposition processes can be modelled by solving the quasi-2D continuity equation of suspended sediment. Neglecting horizontal diffusion, the continuity equation for the *i-th* cell read as follows:

|  |  |
| --- | --- |
|  | (01) |

where:

* is the surface area of cell [m2];
* is the water depth in cell [m];
* is the volumetric sediment concentration in cell [g m-3];
* is the downward vertical flux of fine sediment in cell [g m-2 s-1];
* is the number of neighbour cells
* is the water discharge between cells and [m3 s-1];
* is the sediment concentration in the cell from where the flow is leaving [g m-3].

The downward flux is expressed as:

|  |  |
| --- | --- |
|  | (02) |
|  | (03) |

where:

* is the probability of deposition [adim.];
* is the fall velocity of suspended sediment particles [m s-1];
* is the mean flow velocity in cell [m s-1];
* is the critical mean flow velocity for deposition [m s-1].

It is important to highlight that the hydrodynamic model computes the water depth in the cell. However, flow velocity and discharge are computed for the sections between adjacent cells.

# Numerical Implementation

Equation (01) can be solved using a finite-difference numerical scheme. Here a first-order numerical procedure was adopted (Euler Method). Thus the sediment concentration on a future time-step can be computed as follow:

|  |  |
| --- | --- |
|  | (04) |

where:

* is the current time
* is the time-step used in the output files of the hydrodynamic model [s];
* is a time moment between and that represents the time-average

condition of discharge or sediment concentration;

As the hydrodynamic model can be executed beforehand, the time series of water depth, velocity and discharge are previously available. Thus, it is possible to set in equation (04) resulting in the appearance of and in terms of the right side.

The sediment concentration given by can be either or . If the flow is moving from cell to cell , than , otherwise, if the flow is in the opposite direction, then .

The discharge between cells, , is already known, but its signal depends on its direction. When the flow is leaving cell towards cell , the discharge is negative. If the flow is coming from cell to cell , than the discharge is positive. In this way, the sediment concentration in the cell is always in a product with a negative discharge, while the concentration in cell is always multiplied by a positive discharge.

Therefore, to reorganize equation (04) in a single function to compute we can define:

|  |  |
| --- | --- |
|  | (05) |
|  | (06) |

Hence, the last term of equation (04) becomes:

|  |  |
| --- | --- |
|  | (07) |

With this, we can compute by:

|  |  |
| --- | --- |
|  | (08) |

On each time step is updated using the water velocity data at the same time step. The average velocity, , is calculated from its Cartesian components. Therefore, it is assumed that the domain is made of regular cells, and each cell may have an East, North, West and South face. Denoting the velocity in these respective faces by , , and , we should have:

|  |  |
| --- | --- |
|  | (09) |

With this approach cannot be computed explicitly because equation (06) demands the concentration in the neighbouring cells in the same time step. Therefore an implicit solution was implemented, which is described as follow:

1. Initialize the sediment concentration at . It can be used the concentration in the final step of a previous simulation. Other approach is to set a default input concentration in the input cells and zero for all other cells in the domain.
2. Set a first guess of the sediment concentration in the next time step, , as the same concentration in the actual time step, .
3. Compute for : (*i*) the sum of negative discharges in each cell, ; (*ii*) the mean water velocity, ; (*iii*) the probability of deposition, . With these values at hand, it is possible to compute the denominator of equation (08), let call it , avoiding repeat this computation in the iterative process.
4. Set an initial value for the maximum error, , that is higher than the tolerance, (i.e. and ). Also start an iterator counting, .
5. While :
   1. For each cell in the domain, do:
      1. Skip if the cell is an input cell;
      2. Compute the sum of the products of positive discharges with the sediment concentration in the neighbours cells, , with the guessed values of concentration;
      3. If , compute using equation (08). Otherwise, make to prevent instabilities in the model.
   2. Estimate the maximum error by and increment the iteration counting.
   3. Replace the guess with the estimate of future concentration:
6. If desired, display the values of maximum error and number of iterations to reach the tolerance.
7. Compute the mass of sediment deposited in each cell, , by:
8. Increment the actual time in one time step and go back to step 5. Do this until reach the last time step of the simulation.