FAQ on ShorelineS based on case studies

Project: TKI ShorelineS
Date: 1 October 2023
Author : Bas Huisman
bas.huisman@deltares.nl

How to define the coastline?

- The coastline needs to be specified either as x-y vectors (in S.x_mc and S.y_mc as [1xN] vectors) or in a textfile with the xy-coordinates (in *S.LDBcoastline* with [Nx2] vector).
- Separate coastline segments can be specified, which need to be separated by NaN's in the *S.x_mc/S.y_mc* or *S.LDBcoastline.*
- At most one 'open' coastal element (or line-element) can be used, which should be the first element specified. Other elements should be closed (i.e. islands with begin and end point matching). The first element may also be a closed-element.
- Land and lake segments can be defined. The land-segments are defined clockwise, while lakes should be defined anticlockwise. The lakes should be inside coastal elements (so within an island or landward of an open-coast). The first element should always be a land-segment. So, lakes can only be the second element or later.
- Only cartesian coordinate systems can be used with units in meters.
- Note that the prescribed elements are regridded in the model, which may increase or decrease the resolution depending on the specified grid cell size (*S.ds0*)
- In general, it is preferable to have a rather smooth initial coastline. Especially, if it is a coast with high-angle wave conditions.

How to define the model grid? What is the typical grid cell size?

- Typical grid cell size (*S.ds0*) of 50-100 m
- Larger and smaller grid cell sizes can be used. Coarse grids are suited for large-scale problems where not too much detail is needed. Fine grid cells need to be accompanied by a small timestep and precise wave conditions. It is known that grid resolutions of ~10 meter pose some difficulties for a stable model run.
- Two 'grid regerenation methods' are available. By default the whole grid is re-interpolated when a cell is too small or too large (*S.griddingmethod=2).* A second method has been introduced which only splits or merges these cells that are actually too large or too small (and not redefine all cells), which can be activated using *S.griddingmethod=1*. With this method it is also possible to perform local grid refinement.
- A smoothing factor can be used to re-arrange grid every timestep when griddingmethod==2 is used (*S.smoothfac*). Reasonable values range from 0 to 0.1. It is preferable to not use a smoothing factor (by default *S.smoothfac*=0).
- A spatially varying grid size can be used if griddingmethod 1 is used. This requires the S.ds0 to have multiple grid-sizes at predefined locations.
  For example: S.ds0=[x1,y1,ds1; x2,y2,ds2; ... ; xn,yn,dsn];

How to do local grid refinement?

- With the model the grid can be defined spatial varying with gradual transitions. It is not enforced 1-on-1, but the model aims for generating approximately the specified grid cell size.
- It is required to use the method *S.griddingmethod=1* to allow local grid refinement.

- Instead of a single-value in *S.ds0* it requires the specification of a *S.ds0* for a couple of x,y locations. The input is as follows : *S.ds0=[x1,y1,ds1; x2,y2,ds2; … ; xi,yidsi]*

How to define begin and end time? What is the typical time step?

- The start time has to be entered as yyyy-mm-dd (e.g. S.reftime='2020-01-01')
- The end of the simulation is added as yyyy-mm-dd (e.g. S.endofsimulation='2040-01-01')
- The variable timestep is used by default. It is automatically computed based on the transport rates. You can set the fraction of this automatic timestep with *S.tc*. So, a *S.tc=0.9* uses a timestep that is 90% of the maximum automatically computed timestep.
- Note that when a variable timestep is used the timestep will always be the minimum of the automatically computed timestep and the DT at which the wave conditions (in the input file) are specified. This ensures that wave conditions in the input file are never skipped.
- For more control, a fixed timestep can be used by setting *S.tc=0* and setting *S.dt* to a value in years (e.g. *S.dt=1/365* to get a one day timestep).
- A typical timestep for a coastline model is 3hrs to 1 day.
- If you use a fixed timestep, then the model aggregates the input wave conditions within the fixed dt and applies this averaged wave condition. So, for a *S.dt=1/365* (of one day) and a time-series that is defined every 3 hours, the model will take the average of all wave condition 8 instances that occur within the considered time step of a day. It will add up the 'energy vectors' of all these wave conditions, take the mean, and translate the 'average energy vector' back to an average condition.

How to define wave conditions?

- Wave conditions can either be specified as a static value, as a time-series or as a wave climate with probability per conditions.
- A static wave height can be specified in *S.Hso* (significant wave height in meters), the wave direction in *S.phiw0* (as wave direction in nautical convention °N) and the wave period in *S.tper* (as peak wave period in seconds). A standard spreading around the mean-wave direction of 90 degrees is used (in *S.spread*) which can be adjusted.
- A single time-series can be specified using *S. WVCfile='filename.wvt'*. The model checks the extension 'wvt' and then knows it is a time-series. The time-series file has the format of four columns with date/time (as a string of yyyymmddHHMM), wave height (*Hs* in meters), wave period (*Tp* in seconds) and wave direction (*Dir* in nautical convention).
- A single wave climate can be specified using *S. WVCfile='filename.wvc'*. The model checks the extension 'wvc' and then knows it is a wave climate. The wave climate file has the format of four columns [Nx4] with wave height (*Hs* in meters), wave period (*Tp* in seconds), wave direction (*Dir* in nautical convention) and probability (as a fraction of 1, or in percentage). The probability of the conditions are scaled with the sum of all the probabilities. It is noted that a wave climate should not be combined with a variable timestep, but with a fixed timestep (i.e. S.tc=0 and S.dt at a value). Since a variable timestep would over-emphasize the low-energy conditions.
- Spatially varying time-series or wave climates can be specified by either using a cell-string as input for the *S.WVCfile* or a 'referencing file'.
    - A cell string {Nx3} can be used to apply the spatially varying wave climate wherein the first column has the names of the time-series/wave-climate files, and columns 2 and 3 respectively the x and y locations of these input wave files. As an example, it may look like *S.WVCfile={'example1.wvt',x1,x1; 'example2.wvt',x2,y2; etc};*
    - Spatially varying time-series or wave climates may also be specified by using a .WVT-file or .WVC-file containing a table with a reference to multiple WVT/WVC-files and their x and y locations as input for the *S.WVCfile*. With the first column in this 'reference WVT/WVC-file' being the names of the time-series/wave-climate files, and columns 2 and 3 respectively the x and y locations of these input wave files. As an example, it may look

like *S.WVCfile='referencing.wvt'*; And the content of the 'referencing.wvt' contains *'example1.wvt',x1,x1* and *'example2.wvt',x2,y2* at the next line etc. It works similar for a wave climate but with WVC-files (using '.wvc' extensions).

- Alternatively, a NetCDF file can be used to provide the nearshore conditions, which is recognized from the '.nc' extension in the *S.WVCfile*. It needs to contain variables 'station_x', 'station_y' for the locations. Time/Date needs to be stored in a 'time' variable, which is in seconds with respect to a refence year on 1 january 1979. Wave properties need to be stored in fields 'point_hm0', 'point_tp' and 'point_wavdir'.
- In general, it is preferable to use nearshore wave conditions at the depth-of-closure (i.e. close to the coast; possible even closer to the shore) which implicitly includes the transformation of the foreland in the wave boundary conditions, as this has been dealt with in the 2D wave computation.
- The selection of wave conditions for the different boundary points has been synchronized, such that for all sections the same randomly drawn wave climate condition is used.
- The seeding number of randomized selection of wave climate conditions can be prescribed in *S.randomseed*, which allows for re-generation of the same random series. By default, this is off, meaning a different seeding number is taken for every new simulation.
- Looping the wave conditions of a timeseries is not possible, this requires some preprocessing.

In what way can I specify the foreshore orientation? And why does it matter?

- The foreshore area just outside the region with the longshore current (i.e. outside the inner depth of-closure) does not react instantaneously to changes at the waterline. And for this region the foreshore orientation is specified separately for the deep water refraction from the offshore (user prescribed point) to the nearshore depth-of-closure (*tdp*). In many cases the disturbance at the coastline is not present at the foreshore, which means that defining the foreshore orientation can have a large influence on the transformed nearshore waves.
- The foreshore orientation of the coastline can be specified using the *S.phif*.
- By default the foreshore orientation is not specified (*S.phif=[]*) which means that the initial coastline is assumed to be representative for the deeper foreshore.
- A fixed value can be prescribed for the foreshore orientation (e.g. *S.phif=270;* in °N describes a foreshore with a sea in the West and land in the East).
- It is also possible to provide a spatially varying foreshore orientation (e.g. *S.phif=[x1,y1,phif1; x2,y2,phif2; etc]*)
- An option is also to use the current orientation of the coast but with some smoothing using a cell-string as input (e.g. *S.phif={'gaussian',7}* which smooths over 7 cells).

What happens with the prescribed wave conditions in the model?

- Wave conditions are interpolated on the transport grid points at every timestep. For this purpose two methods can be chosen for the interpolation method which are:
  - 'S.interpolationmethod='alongshore_mapping' which projects the wave locations on the coast and then interpolates alongshore, which is the most accurate but also the slowest.
  - S.interpolationmethod='weighted_distance' which takes a weighted average of the nearest two wave locations based on the inverse of the distance to these points, which is less accurate but much quicker.
- The prescribed wave conditions are transformed in nearshore direction using snellius for wave refraction, depth-induced breaking (based on *S.gamma*) and shoaling based on the wave-celerity at the considered depth. Both wave height and direction are adjusted for the wave refraction. This is done first for the 'relatively static' foreshore from offshore (o) to the depth-of-closure (tdp) and then for the nearshore to the point-of-breaking (br).

- So, three cross-shore locations are evaluated which are the offshore (or user-defined boundary condition depth, referred to with *o* symbol), the nearshore location at the depth-of-closure (*tdp*) and the point of breaking (*br*).
- The offshore depth is specified as *S.ddeep* and the nearshore depth as *S.dnearshore*. Note that *S.ddeep* is always the depth at the point at which the wave data were derived, whereas the nearshore depth should be estimated based on the depth-of-closure.
- *S.ddeep* (at point of wave output) may in some cases be equal to *S.dnearshore* when nearshore waves are applied to the model.
- The transport rates are computed using the offshore wave conditions for the CERC formulation and Kamphuis (*CERC*, *CERC2*, *KAMP* and *MILH*), and using the waves at point of breaking for all other transport formulations (e.g. *CERC3* or *VR14*).

Which transport formulations are available? What parameters should I specify? And how do I include tide?

- A couple of transport formulations can be chosen. They can be subdivided in formulations that use 1) directly the user-defined offshore wave condition, 2) use nearshore wave conditions at the point-of-breaking as computed by the model and 3) compute the wave transformation and transports over the whole cross-shore profile.
- The formulation using directly the specified (offshore) wave condition by the user (and not the refraction on the foreshore) are:
    - the CERC-formulation (*S.trform='CERC'*) using a simple wave energy approach based on just the wave height (*S.Hso*) and wave direction (*S.phiw0*). *S.b* is a calibration factor for only this CERC formulation.
    - CERC with implicit wave refraction (*S.trform='CERC2'*) assuming a foreshore with same orientation as the coastline. This requires also the porosity (*S.por*), depth-induced breaking parameter (*S.gamma*) and density of the water and sediment (*S.rhow* and *S.rhos* in kg/m^3).
    - Kamphuis formulation (*S.trform='KAMP'*) includes the wave period (*S.tper*), beach slope (*S.tanbeta*) and median grain size (*S.d50* in meters).
    - Mil-Homens (*S.trform='MILH'*)
- Nearshore wave conditions at the point-of-breaking are used for:
    - A modified nearshore CERC-formulation (*S.trform='CERC3'*) which is similar to the regular CERC-formulation, but with the median grain size and adapted nearshore coefficients.
    - The Van Rijn 2015 formulation (S.trform='VR14') is a parameterization of the TRANSPOR2004 formulation (Van Rijn, 2007). It is very much like the Mil-Homens and Kamphuis formulation, but requires also a fraction of swell waves (*S.Pswell*)
- The wave transformation and transports are computed over the whole cross-shore profile for the Soulsby-VanRijn formulation with tide (*S.trform='WAVETIDEPROF'*). This requires not along the specification of the waves at the boundary, but also the bed-friction coefficient (*S.Cf*), a minimum water depth (*S.hmin*) and closure depth for the tidal transport (*S.hclosure*) specifying till what offshore extent the tide-induced bed changes can still be of relevance for the coastline. Furthermore, a set of tidal constituents of the M2 and M4 tides needs to be provided at a number of alongshore locations at depths of about 5 meter. These tidal constituents need to be derived at the considered locations from the water-level time-series of a 2D tide-model that is driven by M2 and M4 tide, and specified in a table in a tide-input file (*S.tidefile*). The input is:
    - x and y location of the stations ('x_stat' and 'y_stat'), which are the first two columns.
    - vertical amplitude of tide components ('eta' in meter) in the 2th and 4th column,
    - longshore gradient of eta ('*detads*') in the 5th and 6th column,
    - phase of vertical tide components ('*phi*' in degrees) in the 7th and 8th column,
    - alongshore wave number of tidal components ('*k*' in radians/meter) in column 9 and 10.
    - mean longshore surface slope ('*surfslope*') in the 11th column.

- All transport formulation can be calibrated using the *S.qscal* parameter. A factor of 1 means no calibration of the transport rates.

What is the relevance of active height? And how should I determine it?

- Active height is used to describe the part of the profile that actively moves with the coastline
- This starts from the inner depth-of-closure and extents towards the dune toe. Or sometimes even includes the whole dune when an erosive coast is concerned.
- The vertical distance between those levels is specified as the active height *(S.d)*.
- The active height is used in the code to translate the accumulation (or erosion) of sediment in m^3/yr in a grid cell (with length ds) to a coastline change in m/yr.
- A spatially varying active height can be specified (S.d=[x1,y1,hactive1; x2,y2,hactive2; etc])
- It is noted that the active profile height and the transport rate (scaling) are co-varying.

Which boundary conditions can be used for open coasts?

- You can specify open and closed boundaries for the start and end of the model in *S.boundary_condition_start* and *S.boundary_condition_end*.
- The options for the boundaries are :
  - closed boundary (='closed' or ={'closed',25000} for a 25 thousand m^3/yr transport)
  - fixed coastline position (='fixed' or 'Neumann')
  - a fixed coastline orientation (='angleconstant' or ={'angleconstant',310} for a fixed 310°N angle at the boundary)
  - a periodic boundary condition where the transport at the start is averaged with that at the end of the grid (='Periodic')
- The coastline change at the boundary grid cells is forced to be perpendicular to the initial orientation of the coastline.

How can I implement regular beach nourishments in the model?

- The nourishments can switched on or off by setting *S.nourish* (with 0 being off, and 1 being on).
- The nourishments can be specified in a '.nor' file in *S.LDBnourish*. This NOR-file contains a table with a line of information for each nourishment. Each nourishment should be defined as:
  - x and y location of the begin of the nourishment (columns 1 and 2)
  - x and y location of the end of the nourishment (columns 3 and 4)
  - start and end time of nourishing in 'yyyymmdd' (columns 5 and 6)
  - volume of the nourishment in m^3 (column 7)
- Alternatively, a polygon may be defined with a polygon file (S.LDBnourish='example.pol' with nourishment locations [Nx2] and multiple nourishments separated by NaN's). Furthermore, the rate of the nourishment (S.nourratefile='example_rate.txt' in m^3/yr), start and end date (S.nourstartfile='example_start.txt'; S.nourendfile='example_end.txt' in 'yyyy-mm-dd') need to be provided. With a line for each of the considered nourishments.
- Sediment from the nourishments is automatically distributed over the considered grid cells.
- Keep in mind that shoreface nourishments are, in reality, not immediately affecting the coastline. The timescale for the feeding of the nourishment may be longer than the construction period, which is under consideration. The feeder effect of shoreface nourishments is likely present in a future release.
- It is noted that nourishment can also be used to prescribe sediment sinks (e.g. at beaches with steep tidal channel slopes) or to add sediment at places with bar-welding.

How can I implement shoreface nourishments in the model? (i.e. underwater berms of sand)

- Shoreface nourishments slowly feed the coast and are therefore not applied instantaneously at the waterline. Instead the sediment is slowly applied on the beaches based on a diffusion coefficient.
- The shoreface nourishments can be switched on using *S.fnourish=1*.
- A diffusion coefficient is either enforced by the user or automatically computed. In order to force a diffusion coefficient the user needs to specify a K per shoreface nourishment (specified as [Nx1] in *S.K*). The actual rate of diffusion is then this factor. The alternative is that the model automatically estimates a suitable diffusion coefficient, which has been trained for the situation in the Netherlands. This automatic evaluation of K requires the user to leave the diffusion coefficient empty by *S.K=[]*. Then the volume, length, grainsize and depth are used to compute the K.
- A file is to be provided wherein the properties of the shoreface nourishments are specified (in *S.fnorfile*) with file-extension .fnor. This .fnor-file contains 1 row per shoreface nourishment, with at each column :
    - Start x-location [m of coordinate system]
    - End x-location [m of coordinate system]
    - Start y-location [m of coordinate system]
    - End y-location [m of coordinate system]
    - Alongshore length of the shoreface nourishment [m]
    - Placement depth [m]
    - Median grain size (d_{50}) [m]
    - Placement moment tstart in [yyyymmdd]
    - The total volume of the shoreface nourishment [m^3]

What should I keep in mind using revetments?

- Revetments can be switched on/off by setting the *S.revet* field to 0 or 1. If not set, then the model will try to detect the use of structures based on other input (e.g. whether there is data in *S.x_revet, S.y_revet* or *S.LDBrevetments*).
- The revetments can be specified in *S.x_revet* and *S.y_revet*. Specify a series of x or y-points [1xN].
- If you want to specify more revetments, then please separate them with a NaN. So, *S.x_revet=[x-values revetment 1 , NaN, x-values revetment 2, NaN, …]*;
- Alternatively, a textfile with the revetments can be specified in *S.LDBrevetments* (as a string of the filename; i.e. instead of using the *S.x_revet* and *S.y_revet*). This textfile should contain a [Nx2] table of the xy-points of the revetments which is separated by NaN's between the revetments.
- The revetments bypass sediment from one side to the other. The amount of sediment coming from updrift and the available sediment at the cell will provide a limit to the sediment transport in the downdrift cell.
- Revetments also have a transition zone (or cross-shore width), wherein they reduce transport. The *S.crit_width* sets the cross-shore width. Below this critical width the transport will be reduced linearly. So, if there if is only half this width of sand present in front of the revetment, then the transport will be halved. Typically, the *S.crit_width* is set to a small value of 5 meter or less.
- It is not possible to describe a slope of the structure, but the crit_width factor may be used to scale transport and bypass.

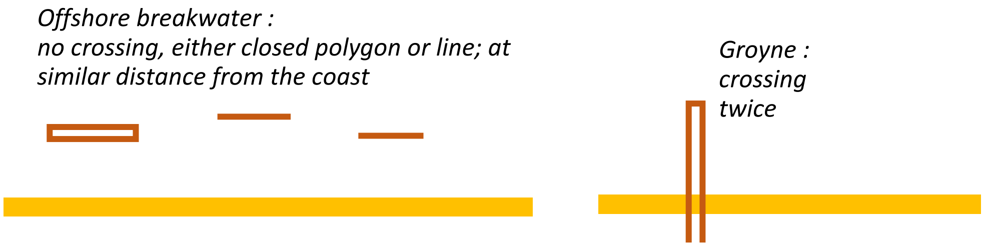
What should I keep in mind using offshore breakwaters? (i.e. detached structures with diffraction)

- Structures (groynes and/or breakwaters) can be switched on/off by setting the *S.struct* field to 0 or 1. If not set, then the model will try to detect the use of structures based on other input (e.g. whether there is data in *S.x_hard, S.y_hard* or *S.LDBstructures*).
- An offshore breakwater (or groyne/breakwater without bypassing) can be specified directly with coordinates or as a text file.
    - Specifying the structures with coordinates can be done in the keywords *S.x_hard* and *S.y_hard*. This requires a [1xN] series of the x and y-points in these keywords. If you want

to specify more offshore breakwaters, then separate them with a NaN. So, *S.x_hard=[x-values offshore breakwater 1 , NaN, x-values offshore breakwater 2, NaN, …]*.

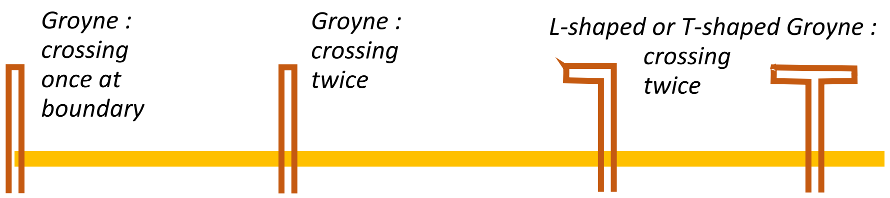
- A reference to a textfile with the xy-coordinates of the hard structures can be given in *S.LDBstructures* (as a string of the filename). This textfile should contain a [Nx2] table of the xy-points of the offshore breakwaters. Different structures should be separated by NaN's. So: *[x and y-values of breakwater 1 ; NaN, NaN; x and y-values of breakwater 2; …]*.

- The model will detect whether a structure is a groyne or an offshore breakwater depending on the crossings it has with the coastline. An offshore breakwater has no crossings with the coastline. These offshore breakwaters should be located offshore at a somewhat similar distance from the coastline, and not be shading each other. If a groyne or breakwater is in front of another structure, then only an additional reduction of the wave energy is applied.



What should I keep in mind using groynes? (i.e. cross-shore structures with bypassing and diffraction)

- A groyne needs to be specified in the *S.x_hard* and *S.y_hard* in a similar way as the offshore breakwater (and in the same field/file). This needs to be a series of x or y-points [1xN]. If you want to specify more groynes, then please separate them with a NaN. So, *S.x_hard=[x-values groyne 1 , NaN, x-values groyne 2, NaN, …]*; Alternatively, a textfile with the hard structures can be specified in *S.LDBstructures*.
- The model will detect whether a structure is a groyne or an offshore breakwater depending on the crossings it has with the coastline. The model will identify the structure as a groyne when it crosses the coast 2 times. Or alternatively, at least 1 time at the boundary of the model. A groyne is basically a sort of a 'staple'. Preferably, the first crossing is at a lower coastline index than the second. It is defined clockwise first from land in seaward direction, then along the positive direction of the shore, and then a crossing back to land.
- As an alternative the groyne can be defined at the boundary of model also. In this case the groyne should also have 'two legs'. One of which crosses the coastline close to the boundary (within a single grid cell distance), while the other leg is just outside the model domain. The model will check if a crossing with this second groyne leg occurs if the model boundaries are extended by a distance of 10x the grid size (ds0). If this is the case, the model will stick the coastline boundary to the groyne edge. It is highly recommended to combine this with a 'closed' boundary. The specified rate of transport at this boundary will be used as the bypassing of the groyne, but please carefully inspect this.



- It is possible to define curved, L-shaped or T-shaped groynes. The main thing is that they do cross the coastline as the regular groynes do. The model will identify the diffraction points depending on the wave condition when wave diffraction is switched on.

- Re-starts of the model are needed to either construct and/or decommission structures over time, which requires an effort from the modeler.

How does sediment bypass at a groyne work?

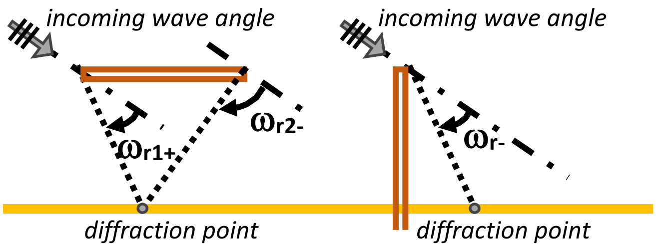
- The bypass of the groyne will depend on the ratio of the 'point of breaking' and the 'depth at the breakwater-tip'.
- The point of breaking is computed by dividing the wave height (*Hs*) by the depth-induced breaking parameter (γ) and scaling it with the parameter *S.Aw*. By default, the model uses an *S.Aw* of 5 when a static climate is used (which is set in *S.Awfixedhs*), which is relevant for an average climate condition, and indicates that the breaking depth can be five times deeper than the average *Hs* that is specified. An *S.Aw* of 1.27 is by default used in case of a time-series, wherein very high wave conditions can be present.
- The depth at the breakwater-tip is estimated using a dean-profile with an extent similar to that of the groyne, which is scaled with the median grain diameter (*S.d50* in meters). The depth at breakwater tip = (1.04+0.086*log(d50))^2 * cross-shore extent^0.67.
- It is possible to include an enhancement factor for the bypassing of groynes. The transport capacity at the tip is enhanced, which effectively means that the coast will not grow to the tip of the groyne but earlier on get to an equilbrium cross-shore extent (where supply is equal to the bypass rate). So, the coast may grow for example just to halfway the length of the groyne. This requires the user to set the *S.bypasscontractionfactor* larger than 1 (i.e. between 1 and 1.4 typically for realistic cases). This factor needs to be calibrated on the local situation.
- Sediment bypass is by default distributed evenly over the downdrift part of the groyne up to the point where the 'shadowed transport' is less than the 'unshadowed transport' would have been. It is possible to distribute the bypass closer to the groyne using the *S.bypassdistribution_power* with a higher value than 1. Two means a quadratic decay away from the groyne instead of linear.

How to use wave diffraction at offshore breakwaters and groynes?

- Wave diffraction can be switched on by using *S.diffraction=1*. The model plots will show tiny grey dots at coastline points with diffraction.
- When diffraction is used an additional term for the gradient in wave energy is resolved in the the 'transport computation' which resembles the effect of residual water-level setup driven circulations.
- The wave height reduction (k_d) due to wave diffraction is described by a linear function that depends on the re-orientation of the incoming wave angle (ω_r), as it is coming from the diffraction point.

$$k_d = H_d/H_i$$

With:
K_d = diffraction coefficient
H_d = diffracted wave height
H_i = incident wave height



- Two types of formulations can be chosen for computing the wave height reduction (*kd*) as a result of diffraction at structures (*S.kdform*), which are the 'Roelvink' and 'Kamphuis' approach. The 'Roelvink' approach provides a somewhat narrower area with diffracted wave energy behind a structure, while the 'Kamphuis' approach has a more diffuse result.

Roelvink : $k_d = 1 - e^{-\left(\frac{0.5}{\omega_x}\right)^4}$

$$\omega_x = \frac{(\omega_r + 90)}{180}$$

Kamphuis (1992) :

$$k_d = 0.69 + 0.008\,\omega_r \quad \text{for } 0° > \omega_r > -90°$$
$$k_d = 0.71 + 0.37\sin(\omega_r) \quad \text{for } 40° > \omega_r > 0°$$
$$k_d = 0.83 + 0.17\sin(\omega_r) \quad \text{for } 90° > \omega_r > 40°$$

- The wave diffraction at an offshore breakwater combines the diffracted waves from both sides of the structure using an addition of the wave energy vectors (see also the explanation of the aggregation of wave conditions). So: $k_{d,obw} = (k_{d1}^2 + k_{d2}^2)^{0.5}$
- With *S.dirspr* the directional spreading can be specified, which affects the influence of diffracted waves behind structures. Note that the model has been derived/tested for a small directional spreading (i.e. swell waves) with a *S.dirspr* of 10 degrees.
- The effect of directional spreading on the reorientation of waves in the shadow zone is determined by the option *S.wdform*. By default the 'Roelvink' approach is used.
    - The 'Roelvink' approach for the *wdform* sets a rotation-offset ($\Delta\omega_{rot}$) that is added to the rotation that the wave needs to make from the diffraction point to the coastline point, and also scales the degree at which the rotation in the shaded zone is accounted for (*rotfac*). The actual wave angle rotation is then the net incoming angle (from diffraction point to coast point) minus the $\Delta\omega_{rot}$ and then times the *rotfac*. The *rotfac* is fixed at 0.8 and the $\Delta\omega_{rot}$ scales from -20° to -35° in a range of directional spreading from 12° to 32°.
    - As an alternative the *'Dabees'* or *'Kamphuis'* method can be used for the directional spreading, which means that the wave-angle offset ($\Delta\omega_{rot}$) is set at 0 and the rotation factor at 1. This is effectively directly using the wave-angle from the diffraction point. Directional spreading is then included through a probabilistic determination of the wave-height reduction value (*kd*) for a range of directional sectors, which each contribute to the local wave height reduction.

How to use wave transmission at offshore breakwaters?

- Wave transmission at offshore breakwaters can be included by setting the *S.transmission* to 1. Wave transmission over groynes is not possible.
- When switched on an additional component is added for transmission at structures, which is combined with the components due to the wave diffraction. The effect of the diffracted waves is scaled back depending on the degree of wave transmission, scaling diffracted waves back to 0% when the transmission is 100%.
- The wave transmission can be computed with the formulations of 'd Angremont' (*S.transmform=*'angr' = default), 'Van Gent' (*S.transmform='*gent') or Seabrook & Hall (*S.transmform=*'seabrhall').
- By default the routines use a wave angle of the transmitted wave that is influenced the local structure (default : *S.transmdir=1*). This 'influence of the structure on the wave-angle' scales with the rate of transmission. For a transmission of 0% it uses a wave-angle that is perpendicular to the orientation of the offshore breakwater, while it uses fully the undisturbed wave-angle when the transmission is 100%. As an alternative, the undisturbed wave direction can be used always for the transmitted waves over the breakwater by setting *S.transmdir* to 0.
- For the wave transmission it is relevant to set the relevant parameters of the structure:
    - The depth at which the offshore breakwater is specified is given in meters in *S.transmbwdepth*, with a positive value downward (e.g. 8 m).
    - The crest height of the breakwater is specified in meters w.r.t. MSL in *S.transmcrestheight*, with a positive value upward, above MSL (so, -0.75 means 75 cm below MSL).

- The slope of the offshore breakwater is defined in *S.transmslope* as the ratio of the vertical change over the horizontal distance (e.g. a value of 1/3 is often used).
- The width of the breakwater crest is specified in meters in *S.transmcrestwidth*.
- The median stone size of the breakwater can be set with *S.transmd50* (default is 1 meter).
- Water-levels used for the computation of the wave transmission at the offshore breakwater water can be specified as a time-series in a text-file (*S.Watfile*). The water levels time series file has two columns [Nx2] with date/time in the first column (in 'yyyymmddHHMM') and waterlevel the second column (in meter relative to MSL). If water-levels are not specified, a fixed value of 0 meter is used for the water level.

How can you specify a climate impact?

- Sea level rise can be a constant rate (e.g. *S.ccSLR*=0.002 m/yr) or a table is used in *S.ccSLR* with the absolute sea level against time [Nx2]. Wherein 'time in datenum format' and 'sea level with respect to initial situation' are specified. Note that the rates per year are then computed automatically. The S.tanbeta is used as 'slope angle' for the Bruun coastal retreat rule.
- A climate impacted increase of the wave height can be a constant rate per year (e.g. *S.ccHS*=0.001, which is +0.1% increase in HS per year). Or alternatively, a table is used in *S.ccHS* with a time varying wave height chane [Nx2]. Wherein the first column has the 'time in datenum format' and the second column the 'relative change in wave height w.r.t. initial situation'.
- The climate impacted change in wave direction can be a constant rate (e.g. *S.ccDIR*=0.05 °/yr) or a table with the date/time and 'wave direction change with respect to the initial situation' [Nx2]. Wherein 'time in datenum format' and 'relative change in wave direction w.r.t. initial situation' as a # degrees.

How does the overwash function work?

- Overwash can take place if the width of a barrier (in the direction of the incoming waves) is less than the defined minimum spit width (*S.spit_width*). By default a width of 50 meter is used.
- Note that during oblique incident waves it is less likely that overwash takes place as the distance over the spit is larger in the direction of the obliquely incident wave.
- Sediment from the seaward face is moved to the backside of the barrier.
- The active height at the seaward and backward side of the barrier can be set separately. This active height is the combination of the berm (or barrier) height (*S.Bheight*) and the closure-depth at the considered side (respectively *S.Dsf* at the seaward side and *S.Dbb* at the backside). So, more sediment becomes available from retreat at the seaward side than is needed to accrete the backward side of the barrier in case *S.dbb* is smaller than *S.Dsf*.
- The rate of overwash is determined by a timescale factor (*S.owtimescale* in years). The barrier width will linearly develop towards the equilibrium width over the provided timescale/period.

How to include dune development in my model?

- Dune development is activated with the keyword *S.dune=1*
- A dune-timestep (*S.dtdune*) can be set for the aeolian and dune erosion computations which is smaller than the coastline time step. The *S.dtdune* is defined as a fraction of a year. Thus, allowing for erosion processes at smaller time scales. If the S.dtdune is not set, or left empty, then the minimum timestep of the timeseries of the water-level timeseries (WATfile), wind timeseries (WNDfile) and/or offshore wave timeseries (from WVDfile) is used. The timestep for the dunes can never be larger than the timestep for the coastline.
- The properties of the dunes then need to be specified. This can be done using a text-file with a number of typical dune properties along the coast, which is pointed to in *S.LDBdune.* This input file contains a [Nx5] matrix with at each line the local properties of the dunes:
  column 1 : x position of the specified dune properties

column 2 : y position of the specified dune properties
column 3 : berm width (or beach width) from waterline to dune foot (in meter)
column 4 : height of the dune foot (w.r.t. MSL)
column 5 : dune crest elevation (w.r.t. MSL)

- Instead of the 'S.LDBdune' it is possible to specify above 5 parameters seperately (if S.LDBdune='';). Then the model will check for S.xdune, S.ydune, S.Wberm, S.Dfelev, S.Dcelev.
- The following parameters (with default values) are relevant for dune erosion and dune growth.
  S.kf=0.02; % Friction coefficient
  S.Cs=5e-4; % Impact coefficient waves based on thesis M. Ghonim || very sensitive
  S.d50r=2.5e-4; % Median reference grain size
  S.rhoa=1.225; % Air density []
  S.duneAw=0.1; % Coefficient (Bagnold, 1937)
  S.Kw=4.2; % Empirical coefficient (Sherman et al. 2013)
  S.k=0.41; % Von Karman's coefficient
  S.segmaw=0.1; % fraction of the fetch used for the aeolian transport computation
- It is noted that the difference between the 'dune crest' and 'dune foot' defines the active height of the dunes. In many cases the 'active height' of the coastline should be reduced when dunes are used in the model, as the dunes have their own active height.
- Dune growth is driven by wind forcing.
- Dune erosion takes place only when the water level exceeds the dune foot level. This means that both the tide, the run-up, surge water levels play an important role. The run-up in the model is computed on the basis of the (offshore) wave height and beach slope (using berm width). You can select the run-up formulation with *S.runupform*. This can be:
    - 'Stockdon'; (default preferred choice)
    - 'Larson'; (for steep/narrow beaches and if no surge info available; gives higher runup)
    - 'Ghonim'; (similar in formulation to Larson, but with beach width included, and therefore for wider beaches, run-up levels are more like Stockdon)
- This often requires the specification of time-series of the wind, water-levels and offshore wave conditions as used for the surge at the dune foot. The following input conditions can be specified.
  - A wind time-series file or static wind forcing to be specified. The wind time-series can be specified in *S.WNDfile* and is a text-file with [Nx3] format with date/time in 'yyyymmddHHMM' and wind velocity [in m/s] and direction [in degree North]. Alternatively, a static wind climate can be specified with the wind velocity (*S.uz)*, wind direction (*S.phiwind0*) and height at which wind is defined (*S.z*). Furthermore, a wind drag coefficient needs to be specified (*S.Cd*).
  - Surge levels (which need to include also the tide level) in the model are specified using a time-series of waterlevels in a text-file (*S.WATfile*) or as a fixed value (*S.SWL0*). The water level time series file has two columns [Nx2] with date/time in the first column (in 'yyyymmddHHMM') and waterlevel the second column (in meter relative to MSL).
  - The wave heights for the run-up formulation can either be directly used from the wave condition file or specified as offshore wave conditions (dedicated for the run-up) in a separate file in *S.WVDfile* (with WVD-file extension). This file is a text-file with [Nx4] format with date/time in the first column as 'yyyymmddHHMM' and wave height, period and direction in the other columns.
  - It is possible to specify spatially varying offshore wave conditions, wind data and water-level time-series by using either a {Nx3} cell wherein the first column has the names of the time-series/wave-climate files, and columns 2 and 3 respectively the x and y locations. Alternatively, multiple condition files with their locations can be linked to from a 'referencing file' (with the .WAT, .WVD and .WND extensions). With the first column in

this 'reference WVT/WVC-file' being the names of the time-series/wave-climate files, and columns 2 and 3 respectively the x and y locations of these input wave files. This works in a similar way as for the spatially varying WVC files.
  - Note that the offshore wave conditions (WVDfile), wind data file (WNDfile) and water-level file (WATfile) should have the same time instances.

Why do I get high-angle instabilities in the model? When are they physical? And how to set the upwind properties?

- High-angle instabilities may play a role when waves from very oblique angles approach the coast (i.e. more than 40 degrees w.r.t. coastline). In those cases, small perturbations tend to grow, because there is too little wave energy at the downdrift side of a coastal perturbation to transport the sediment away. This is a physical phenomenon and can cause spits to grow or induce shoreline undulations.
- The model does, however, cap these instabilities by recognizing the locations with transitions from low to high-angle incidence. And correct the transport there (i.e. use a maximized transport in the points directly downdrift). In this way spits may grow, as can be seen for the Sand Motor and natural flying spits in Namibia.
- It is advised to use *S.twopoints=1* to spread the sediment at high-angle transitions, which covers also the second downdrift cell to obtain a smooth coastline.
- In addition, the *S.maxangle* can be adjusted to limit the angle change between grid cells. By default *S.maxangle=60°*.

What options are there to make the model stable with small grid sizes?

- A 'relaxation distance' can be prescribed which accounts for the inertia of the flow. The current and subsequent transport will not immediately stop when the forcing quickly drops. By setting the *S.relaxationdistance* (to a length-scale in meters) it takes some space for the transports to decelerate which effectively spreads the sediment over a slightly larger area. This is especially relevant for small grid sizes (less than 50 meter) and/or when situations with high wave-angles are modelled (e.g. at spits).
- A smoothing factor can be used to re-arrange grid every timestep when griddingmethod==2 is used (*S.smoothfac*). Please see the earlier section on the grid definition.
- Smoothing can be applied on the coastline orientation that is used for the computation of the wave transformation from the nearshore depth (tdp) to the point of breaking (br) using the *S.smoothrefrac*. A value of 0 means no averaging, while a value of 1 means full averaging with the neighboring cells. This is especially relevant for small grids where the local orientation can quite easily change, which results in feedback on the waves and an undesired large impact (possibly reduction) on the transport. In those cases it is better to average the orientation of the coastline with the neighboring cells (e.g *S.smoothrefrac*=0.5).

How to visualize the coastline and wave conditions used in the model?

- By default the average waves as prescribed on the model are plotted (at all defined locations) with the offshore wave (black arrow), nearshore wave (green arrow) and wave at point of breaking (brown arrow).
- You can use the *S.plotHS* or *S.plotDIR* to show more detail of the wave height or wave direction at the coastline. *S.plotHS=6* will show you a value and arrow of the wave height at every 6 grid cells. While the *S.plotDIR=6* will show you an arrow of the waves at the nearshore depth *(tdp)* and point of breaking *(br)* respectively with a green and brown arrow. Similarly, the *S.plotQS* will show the transport rates along the coast.
- The interval of plotting to screen can be set with *S.plotinterval* which contains the interval in number of timesteps (so, *S.plotinterval=2* means every two timesteps).

- A time-interval can be set for exporting the figures (*S.fignryear*) which has the number of occasions per year that a figure is written to file. Note that this cannot be more often than the plotinterval.
- XY-limits of the plot can be set with *S.xlimits* and *S.ylimits* ([1x2] of min/max coordinate in m)
- The position of the wave arrows and its scale can be set with *S.XYwave* ([1x2] coordinates).
- An offset can be extracted from the x and y-coordinates in the plot with *S.XYoffset* [1x2]
- The output dir of the plots and the data is placed in *S.outputdir* (e.g. 'Output\')
- The background fill of the coast can be switched on using *S.usefill* = 1 (or off when it is 0)
  For complex coasts the *S.usefillpoints* can be set to a value larger than 0 to obtain a nice landfill behind the specified open coast.
- S.fastplot=1 creates debug plots, which are made very quickly. While more detailed/slower plots can be made using S.fastplot=0.
- In order to speed up the computation it may be decided to make the plot invisible by setting *S.plotvisible=0*. The plots will still be written to jpg-files in the output directory.
- With *S.plotUPW=1* it is possible to plot the locations where the high-angle correction is applied, which show up as green and red squares.

How to plot the coastline and dune change over time at predefined profile locations?

- The model can make plots of the coastline and dune foot positions (i.e. cross-shore) of at cross-sections that are specified in 'xyprofile'. The syntax is : *S.xyprofiles=[x1,y1; x2,y2; xi,yi; …]*

How to handle the output data of the model?

- An output file ('output.mat') will be exported by the model to the specified output directory (*S.outputdir*).
- The time-interval for the output mat-file can be set in *S.storageinterval* which stores the model data at this predefined interval (in days).
- There are two methods of exporting data either using direct output (O-structure) or data that is reinterpolated to a grid (P-structure). Both are written to the 'Output.mat':
  - The O-structure stores the raw information from the computation, which means that variables are defined at grids which may change in size over time. Also note that the O-struture stores the instanteneous data, and not the mean value since the previous timestep. This means the calculation of the cumulative transport can only be done when sufficient output timesteps are used.
  - The P-structure is a re-interpolation of the data on a user specified (curvi)-linear grid. The user can specify this output grid in the "S.xyout" variable. The grid will be fixed in time and space. The model will make a projection of the 'coastline position', 'wave conditions' and 'transports' on this grid. The coastline position (xc and yc) will be exported, which provide the crossing point of the coastline in ShorelineS, at the output-moment, and the 'normal' of the gridline at each output grid cell. Also, the distance from the output grid to the crossing point of the 'normal' with the coastline will be stored (zg). An array will be stored of variables in the P-structure with the coastlines in the vertical and time on the horizontal axis. When applying multiple grids, the P structure will have a dimension of P(gridnumber).
    A mean value is computed of the wave conditions (Hs, Tp and PHI) and sediment transport (QS and QSmax) rates over the period of the output timestep. So, it is not just instanteneous output like the O-structure!
    The new input variable "S.xyout" can be defined in 2 ways:
    - Specify two coordinates as [x1,y1,x2,y2]. A linear grid is made with ds0 spacing. An additional line can be added to the matrix representing a new grid (e.g. [2x4]). So, multiple linear grids can be made.

    - Specify a grid as a cell {} with [Nx2] grids of xy-coordinates. It is used 1 on 1 without interpolation of the gridsize. You can specify multiple grids by adding more cells. For example: {[5x2],[12x2],...}

- Users did experience a slowdown of the model during the run, which relates to writing the output data. This needs to be considered further. Writing files that can slowly grow over time (e.g. NetCDF) may be a solution. But this is still not yet worked out.

References

Kamphuis, W.J., 1991. Alongshore Sediment Transport Rates. J. Waterway, Port, Coastal and Ocean Eng. ASCE, Vol 117, pp624-640.

Kamphuis, J.W., 1992 Computation of coastal morphology (ICCE 1992, Venice ?)

Kamphuis, J.W., 2000: Introduction to Coastal Engineering and Management. Advanced series of Ocean Engineering – Vol. 16.

Wiegel, R.L.,1962 Diffraction of waves by a semi-infinite breakwater, Journal of the Hydraulic Division, Vol. 88, No. HY1, pp. 27-44

Goda, Y., T. Takayama, Y. Suzuki, 1978 Diffraction diagrams for directional random waves Proc. 16th ICCE, ASCE, pp. 628-650

Goda, Y., 1985. Random seas and design of maritime structures, University of Tokyo Press

Shore Protection Manual, 1984