Keywords

Coastline change,

Organizations

Deltares, IHE Delft Institute of Water Education

Authors

Dano Roelvink, Bas Huisman

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Summary  ShorelineS is an open-source numerical model for coastal planform evolution, based on one-dimensional equations for alongshore sediment transport and mass conservation, on a free-form grid that may consist of several coastal sections represented as strings of coastline points. The coastline sections can evolve freely and may include undulations, islands, spits, salient and tombolos; the sections may influence each other through shadowing and may merge or split. A range of hard elements or coastal structures are supported, such as groynes, T-groynes, offshore breakwaters and revetments; processes simulated include bypassing and diffraction. A range of longshore (bulk) transport formulas is available and soft engineering measures such as beach- and nearshore nourishments can be simulated.   |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | --- | | Version | Date | Author | Initials | Review | Initials | Approval | Initials | | 0.1 | 24/11 | Dano Roelvink |  |  |  |  |  | |  |  |  |  |  |  |  |  | |  |  |  |  |  |  |  |  |   State  draft |

Contents

1 Introduction 4

1.1 Description of the development of ShorelineS 4

1.2 Scope 4

1.3 Readers guide 4

1.4 Technical reference version and revisions 4

2 Processes and model formulation 6

2.1 Introduction 6

2.2 Basic equation 6

2.3 Transport formulations 7

2.4 Numerical implementation 8

2.5 Coastline evolution 11

2.6 Dynamic profile concept 12

2.7 High-angle instability 12

2.8 Barrier or spit overwash 12

2.9 Merging and splitting 13

2.10 Treatment of groynes 13

2.11 Revetments 15

3 Input description 16

3.1 Main script or text input file 16

3.2 Definition of coastline(s) 17

3.3 Model grid 17

3.4 Time frame 18

3.5 Wave conditions 18

3.5.1 Static condition 18

3.5.2 Single time series 19

3.6 Foreshore orientation 19

3.7 Mapping of wave conditions to the shoreline 20

3.8 Transport formulations 20

3.9 Active profile height 21

3.10 Coastline boundary conditions 22

3.11 Nourishments 22

3.12 Shoreface nourishments 22

3.13 Revetments 23

3.14 Offshore breakwaters 23

3.15 Groynes 24

3.16 Dune interaction 25

3.17 Overwashing 26

3.18 Channel migration 27

3.19 High-angle instabilities 27

3.20 Climate change 27

3.21 Visualization of coastline and wave conditions 27

3.22 Output data 28

**Appendices**

**No table of contents entries found.**

# Introduction

## Description of the development of ShorelineS

## Scope

This is a technical reference guide which provides a description of the physics behind the many functionalities of the model, the different model settings and the numerical implementation. For the latest news on ShorelineS courses and releases, or if you have questions, visit our website:

## Readers guide

To make this guide more accessible we briefly describe the contents of each chapter and the appendix.

## Technical reference version and revisions

# Processes and model formulation

## Introduction

The description of coastlines in ShorelineS is of strings of grid points (see Figure 1) that can move around, expand and shrink freely. The coastline points are assumed to be representative of the movement of the active coastal profile, and hence are situated at the MSL contour. The model can have multiple sections which may be closed (islands, lagoons). Sections can develop spits and other features and they may break up or merge as the simulation continues.

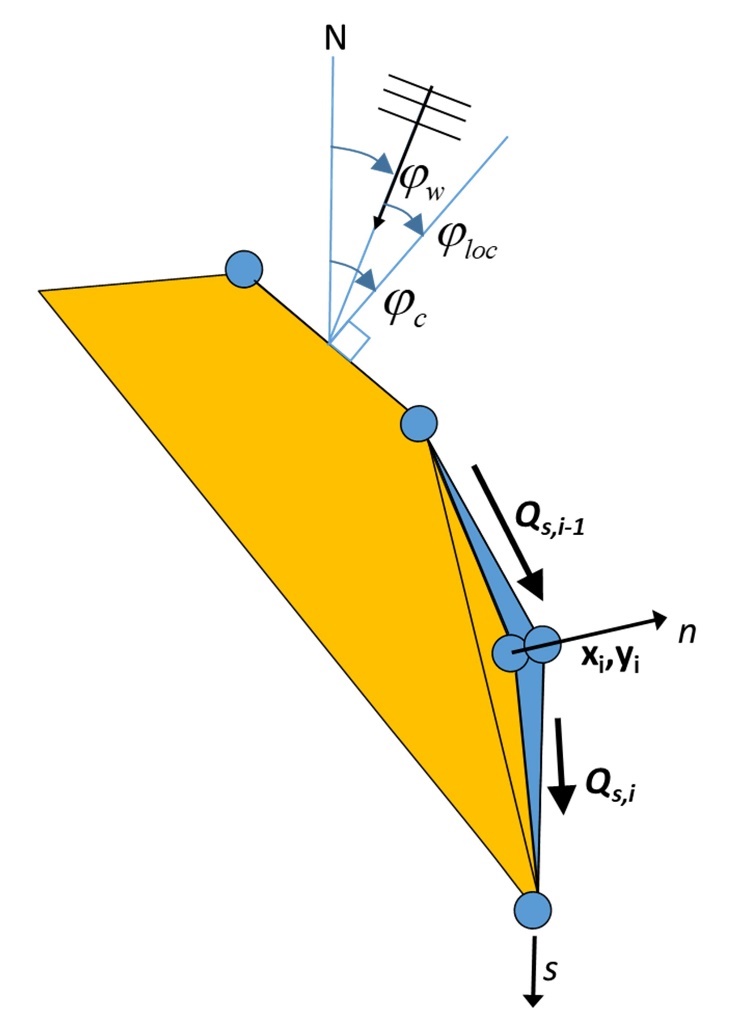


Figure 1. Coastline-following coordinate system and definition of wave and coast angles.  is the orientation of the shore normal with respect to North; is the angle of incidence of the waves with respect to North and the local angle between waves and coast, defined as .

## Basic equation

The basic equation for the updating of the coastline position is based on the conservation of sediment:



where *n* is the cross-shore coordinate, *s* the longshore coordinate, *t* is time, *Dc* is the active profile height, *Qs* is the longshore transport (m3/yr), tan *β* is the average profile slope between the dune or barrier crest and the depth of closure, *RSLR* is the relative sea-level rise (m/yr) and *qi* is the source/sink term (m3/m/yr) due to cross-shore transport, overwashing, nourishments, sand mining and exchanges with rivers and tidal inlets.

## Transport formulations

The coastline changes are driven by wave-driven longshore transport, which is computed using a choice of formulations, which can be calibrated to match the local transport rates. The formulations listed in Table 1 have been implemented. The definitions of the angles are as in Figure 1.

CERC1 and CERC2 are defined in terms of the offshore wave angle, and CERC3 and KAMP are defined in terms of the breaking wave angle. However, in all cases the transport follows a shape rather similar to CERC1 when plotted against the deep water wave angle, with a maximum occurring at an offshore angle of 40° to 45° from wave incidence.

CERC1 is the simplest formula and is mainly meant for illustrating the principles of the behavior of the coastline model. CERC2 is derived from the official CERC formula to formally include the effect of refraction and shoaling. Though its behavior is quite similar to CERC1, it allows for a direct comparison with the Coastal Evolution model that utilises it. The CERC3 and KAMP formulas are widely used in models worldwide such as GENESIS or UNIBEST and again can be useful for intercomparison with such models. CERC1, CERC2 and CERC3 have a single calibration coefficient, whereas the KAMP formula requires, usually uncertain, extra inputs such as beach slope and grain size but has the ambition to be a more accurate, predictive formula.

Table 1 Implemented longshore transport formulations

|  |  |  |
| --- | --- | --- |
| Author | Notation | Formula |
| USACE (1984) (simplified) | CERC1 |  |
| Ashton and Murray (2006) | CERC2 |  |
| USACE (1984) | CERC3 |  |
| Kamphuis (1991) | KAMP |  |

In Table 1, *HS0* and *Hsb* are the significant wave height at the offshore location and point of breaking respectively (m), *T* is the peak wave period (s), *D50* is the median grain diameter (m), *mb* is the mean bed slope (beach slope in the breaking zone), *Φloc* is the relative angle of wave incidence for waves offshore and *Φlocb* is the relative angle of waves at the breaking point; *b* and *K2* are the calibration coefficients of CERC1 and CERC2 formulations respectively, which are computed as :.





where *k* is the default calibration coefficient according to the Shore Protection Manual (USACE, 1984), *ρ* the density of the water (kg/m3), *ρs* the density of the sediment (kg/m3), g the acceleration of gravity (m/s2) and *γ* the breaker criterion.

## Numerical implementation

The ShorelineS model is implemented in Matlab. The flow diagram of the model is depicted in Figure 2. In the following we will describe the procedure point by point.

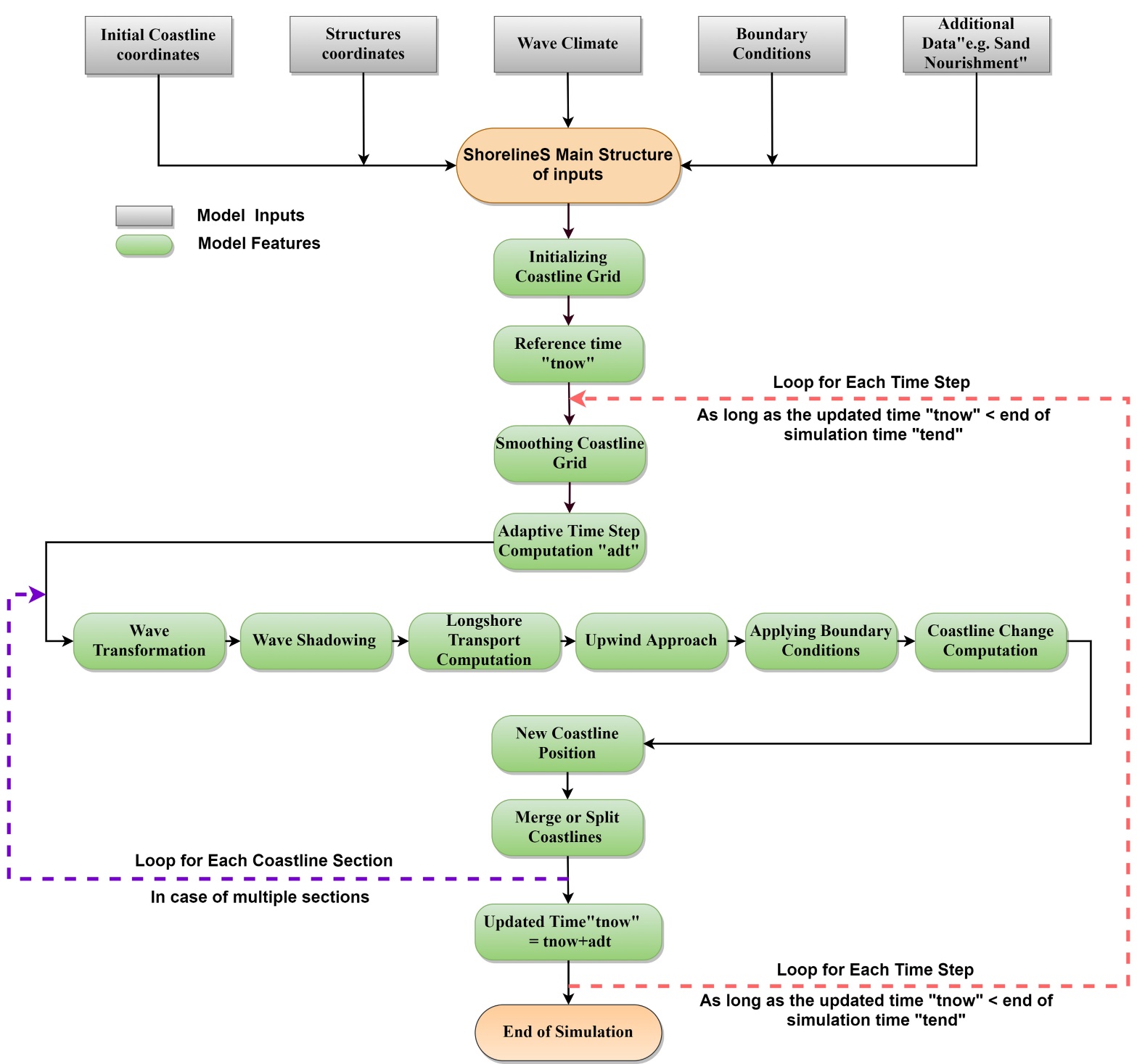


Figure 2. Flow diagram of the ShorelineS model.

The coastline positions are given in two column vectors *xmc* and *ymc*, where the different coast sections are separated by NaN’s. The sea is defined to the left when following the coastline positions. If a section ends at the same coordinates as where it starts, it is treated as a cyclic section and may represent either an island or a closed lagoon. The coordinates may be in any Cartesian (metric) system. Structures are defined in a similar way, as two column vectors where different structures may be defined, separated by NaN’s.

The offshore wave climate can be specified in three ways:

* By means of wave direction and a spreading sector, where a uniform distribution is assumed between the mean wave direction and plus or minus half the spreading sector. For each time step a random wave direction will be chosen from this sector.
* By a wave climate consisting of a number of wave conditions characterized by significant wave height, peak period and mean wave direction, each with equal probability of occurrence. A condition will be chosen randomly for each time step.
* By a time series of these wave conditions, from which the model will interpolate in time.

Various lateral boundary conditions were implemented in the model to represent a variety of coastal situations. For the non-cyclic sections the lateral boundary conditions are specified by controlling the sediment transport rate at the start and end of the boundary, thereby specifying a constant coastline position, a constant coastline orientation or a periodic boundary condition. One type of boundary condition is applied at all open-ended sections, whether existing or newly created. The model detects when a section end point is near the section start point and then always applies cyclic boundary conditions.

Nourishments can be prescribed through a number of polygons within which each nourishment takes place, start and end times, and the total volume of each nourishment. This information is then internally converted into a shoreline accretion rate by dividing the total volume by the time period, the length of coastline within the polygon and the profile height, *Dc*. By the same mechanism sediment discharged by a river can be distributed over a coastline section within a specified polygon. Shoreline recession as a result of relative sea level rise can be specified, e.g., resulting from the Bruun rule (Bruun, 1962), as given by eq. .

All inputs are collected in a single structure *S* that is passed on to the main function ShorelineS. Preparation of the input can be done in a tailor-made script, but ShorelineS and its sub-functions normally do not have to be altered for a specific application. The main function ShorelineS contains default values for all inputs that are not application-dependent.

The cumulative distance *s* along each coast section is computed, and this is then distributed over equidistant longshore grid cells based on a given initial grid size. The *x* and *y* positions of the coastline then are interpolated along *s* to obtain the *x* and *y* positions of the grid points.

In cases where the grid sizes expand (e.g., at the tip of an expanding spit), new grid points are inserted where the grid size exceeds twice the initial prescribed grid size. Where the grid distances shrink (e.g., at an infilling bay or a shrinking spit) grid points are removed when the grid distance becomes less than half the original grid size.

To avoid strong variations in grid size after inserting or extracting grid cells in expanding or shrinking sections, some smoothing of the *s*-grid is applied. The smoothing factor has to be chosen carefully as too much smoothing may lead to a loss of planform area and will tend to straighten out sections that should not move at all. The smoothing formulation applied is a simple 3-point smoothing according to:



where *f* is a smoothing factor, with default value of 0.1. Smoothing can lead to losses in the sediment balance and in situations where this is critical a value closer to zero is advised.

The local wave angle is estimated through the wave transformation from deep water to the nearshore using Snell’s law of refraction and from the nearshore to the breaking line using the equations of van Rijn (2014). The refraction from deep water to the toe of the dynamic profile can be done based on the assumption of parallel offshore depth contours, or using a 2D refraction model to provide alongshore-varying wave conditions.

Some parts of the coastline might be sheltered by structures or other parts (sections) of the coast. Hard structures or rocky shores are represented by an arbitrary number of polylines, which shield waves and block longshore transport where they cross a coastline. Thus, sea walls, hard rocks and headlands can represent supply-limited situations where the transport is determined by the updrift sand supply and ‘plugs’ of sand are bypassed. The waves at any location can be shielded by other coast sections or hard structures, see Figure SI01. This approach is valid when the scale of the structures is much larger than the wave length; if this is not the case, diffraction can be activated using different approximations (Elghandour, 2018).

Given the local wave angle with respect to the coast normal and the refracted wave conditions (or deep water wave directions in the case of the CERC1 and CERC2 formulas) the longshore transport can be computed at each transport point between two adjacent coastline points. At present, a choice of formulations as listed in Table 1 is available to be used.

## Coastline evolution

At each point the local direction of the coast is determined from the two adjacent points (as a reference line), then the longshore transport is calculated for each segment. The difference leads the points to build out or to shrink. The mass conservation equation is solved using a staggered forward time–central space explicit scheme (see Figure 1):



where *j* is the time step index, is the time step (yr), *i* is thepoint/node index and *Li* is the length of the considered grid element computed from and *xi* and *yi* are the Cartesian coordinates of point *i*. From the normal displacement it follows that the change in position of point *i*  then becomes:



The scheme can be shown to be conserving the land area. Since an explicit scheme is applied, the time step is limited by the following criterion (Vitousek & Barnard, 2015):



where the diffusivity is related to the maximum gradient of the sediment transport with respect to the wave angle relative to the coast, which can be approximated by:



where *Qmax* is the maximum transport rate in the model.

Therefore the following is obtained:

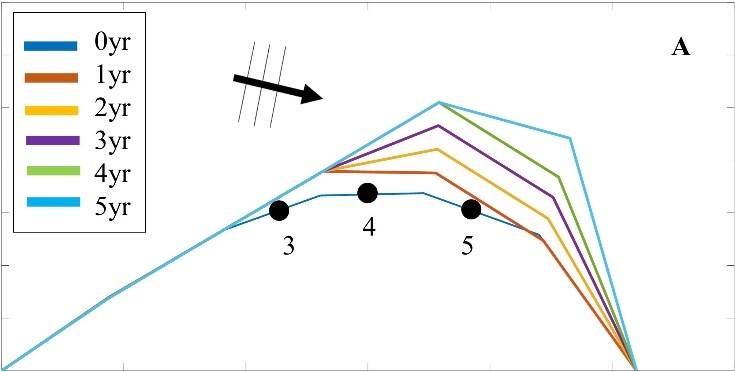


This criterion can be restrictive for small grid sizes (e.g. less than 100m). Stability is, however, guaranteed through this adaptive timestep.

## Dynamic profile concept

## High-angle instability

A special treatment takes care of so-called high-angle instability (Ashton et al., 2001), which allows spits to develop. In cases where the local angle exceeds the critical angle on one side and is less than the critical angle at the updrift side, the transport at the downdrift point is set to the maximum transport (or the angle is set to the critical angle). Figure 3 illustrates the effect of this treatment, where a central scheme would lead to unstable behavior, the local upwind treatment ensures a smooth development into a spit. The physics in the model is the same as in Ashton et al. (2001, 2016), and Ashton and Murray (2006), and therefore it inherits most of the behavior of their Coastal Evolution Model. The novelty in ShorelineS is that it achieves the same behavior with a vector-based rather than a grid-based approach. This is more elegant and more efficient, especially when large areas need to be covered.



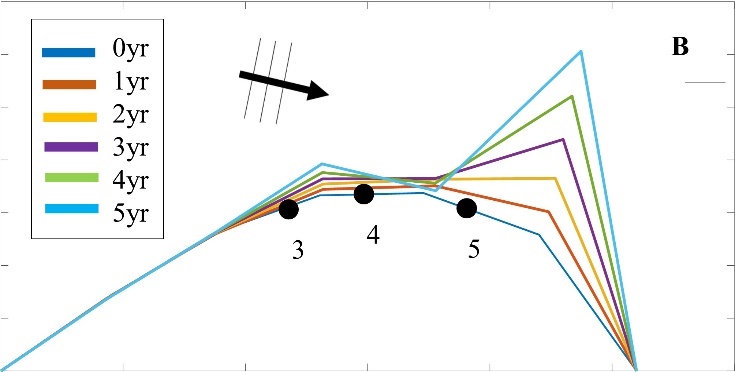


Figure 3. Example of high-angle instability with standard central scheme (A) and upwind scheme (B).

## Barrier or spit overwash

For simulating barriers that already exist or that are in the form of developed spits due to high wave angle instability, it was necessary to represent the overwash process as it maintains the width of the barrier to a certain limit (Leatherman, 1979).

(Ashton & Murray, 2006) introduced the physical process of overwash by assuming a minimum barrier width such that sediment eroded from the seaward side is deposited on the landward side. By simultaneously retreating the seaward and landward sides of a section narrower than the specified critical width, the retreating section creates a longshore transport gradient that tends to fill it up; thus, the retreating helps maintain the width.

A similar concept was implemented in ShorelineS in a simple approach for treating the barrier width. At each time step, the model checks the local barrier width at each point/node, measured in the incident wave direction. If the barrier is narrower than the critical width, then overwash occurs. The overwash process moves the landward point a distance equal to the difference between the actual width and the critical width. Such a distance is not allowed to exceed a given percentage (e.g. 10%) of the local spatial discretization distance of the grid per time step to avoid discretization artefacts. Then the model looks for the closest node on the seaward side to erode it by the same amount (Figure SI02). A possible refinement is, as in Ashton and Murray (2006), to assume different profile depths on the seaward and landward sides, as is logical in some settings, e.g., for the case of an eroding barrier island. In this case the landward extension would be larger than the erosion on the seaward side.

## Merging and splitting

One of the advantages of the ShorelineS model is that it can simulate multiple coastal sections at the same time, and these sections can affect each other by shielding the waves. Small parts of the coast are allowed to split and migrate as the spits are growing and in some cases break up and migrate as a small island. An example of the splitting procedure is shown in Figure SI03. Such splitting typically happens when the seaward side of a section erodes by more than the overwashing process allows for or when the latter is not activated. The numbering is indicated to show how the grid cell connections change after the splitting procedure: from one continuous coastline section to two separately numbered sections.

If two sections intersect, they may merge into one section as the simulation continues, as is illustrated in Figure SI04. Such merging typically happens due to shoreward migration or extension of a spit towards the mainland coast. Again, the numbering is included to indicate how the separate spit and mainland coast sections are now joined at the seaward side as a continuous coastline numbered 12-20 and a lagoon numbered 1-10.

## Treatment of groynes

Groynes can be treated simply as any structure crossing the coastline, where the transport at the transport point closest to the intersection between the structure polyline and the coastline is set to zero. However, such a treatment does not give a very accurate representation of the groyne position and local coastline evolution, and does not account for bypassing in a smooth way. Therefore, a more eleborate treatment was presented in Ghonim (2019), which is summarized as follows. First, additional grid points exactly on either side of each groyne are introduced. Second, the local coastline position at either side of the groyne is forced to move along the groyne. Third, bypassing and transmission are accounted for, according to the following mechanisms.

Bypassing can be simulated in two ways, either as starting only when the updrift accretion has reached the tip of the groyne, or gradually increasing if the depth at the tip of the groyne is less than the depth of active transport. The first approach follows the considerations of , assuming a fully impermeable structure, such as a groyne with complete blockage of the longshore transport. Sand bypassing takes place only when the groyne is filled with sand. Based on that, the longshore sediment transport is set to zero at the structure and the sand bypassing factor (*BPF*) also is set to zero from the start of the simulation until the moment when the sediment reaches the tip of the groyne. Then, the bypassing factor is set to its maximum value (*BPF*=1), which means that all sediment bypasses the groyne’s tip and moves towards its downdrift side. In that case the lateral boundary condition at grid point *i* (see Figure SI05), which is located at the groyne representing the bypassed volume can be expressed as:



where *QSi* is the longshore transport at grid point *i*. There were many options for how the bypassed sediment should be distributed downdrift of the groyne. The most appropriate distribution of the bypassed sediment, in line with the expected flow pattern around the groyne, which attaches roughly at the end of the sheltered area, is to pass all the bypassed sediment at the last sheltered grid point *ilast* and to leave the sheltered area untouched. To do so numerically, the lateral boundary conditions at the downdrift side of the groyne are set as follows:



Eq. (11) ensures that only the last sheltered grid point obtains all the bypassed sediment and equal signs indicate that there is no sediment transport gradient from the grid point *i* to the last sheltered grid point *ilast*. This approach keeps the sheltered grid points fixed in their positions except for the last one, which gives a transport gradient to its following grid point.

That this treatment is more realistic than the classical Pelnard-Considère solution where an erosion peak at the downdrift end of the groyne is assumed follows from many examples worldwide, where the erosion peak is rarely found right next to the groyne but always some distance downdrift, due to the wave sheltering and recirculation in this area. An example is shown in Figure SI06, for a groyne field at Eastbourne, UK.

The second approach (Larson et al., 1987) assumes that sand bypassing does not take place only when the groyne is totally filled with sand, but it may take place just after the construction of the groyne. While sand moves along the coastline, it is influenced by the presence of the shore-normal structures, such as groynes and the response of the coastline to those structures varies for different locations and different types of structures. The main parameters that influence the response of the shoreline at the structure are the structure permeability and the bypassing ratio, which is the ratio between the water depth at the head of the structure *Ds* and the water depth of the active longshore transport *DLT*. The bypassing ratio varies between 0 and 1 (Hanson & Kraus, 2011).

Sand bypassing occurs at the seaward end of the groyne as long as *Ds* is less than *DLT*. The depth of the active longshore transport is similar to the depth of the highest 1/10 waves at the updrift side of the structure (Hanson, 1989), and represents the time-dependent depth for longshore sediment transport, which is often less than closure depth *Dc*, and can be estimated as:



where *Aw* = 1.27, a factor that converts the 1/10 highest wave height to significant wave height [-]; *γ* is the breaker index, the ratio between wave height to wave depth at breaking line [-] and *(H1/3)b* is the significant wave height at the line of breaking [m].

Based on the assumption of equilibrium profile shape (Dean, 1991), the water depth at the structure’s head *Ds* can be determined as:



where *Ap* is the sediment scale parameter [m1/3] and *ystr* is the distance from the structure’s head to the nearest point of the coastline [m]. In that case, the bypassing factor (*BPF*) is estimated based on the following equation:



and the bypassing volume increases until reaching its maximum value when the groyne is filled with sediment [*BPF* =1]. The lateral boundary conditions at the groyne are otherwise equal to those for the first approach, as given by Eqs. (6) and (7).

## Revetments

The basic assumption in our approach is that the longshore transport is reduced when the width of the beach in front of the revetment is reduced; we assume a linear decrease when the width in front of the revetment, , is less than the critical width, , as illustrated in Figure 1.



Figure 1 Decrease of longshore transport, Qs, as function of the beach width.

The basic assumption in our approach is that the longshore transport is reduced when the width of the beach in front of the revetment is reduced; we assume a linear decrease when the width in front of the revetment, , is less than the critical width, , as illustrated in Figure 1.

In order to avoid that the coastline retreats behind the revetment, while conserving mass, we limit the flux at each transport point (at the edge of each coastline cell) to the volume of that cell divided by the timestep:



where is the cell size, *d* the profile height and  the time step, which has been determined based on the transports and grid sizes throughout the model domain.

Finally, in order to avoid gaps between the revetment and the coastline, the coastline is shifted to the revetment where necessary, i.e. when the revetment is initially seaward of the coastline.

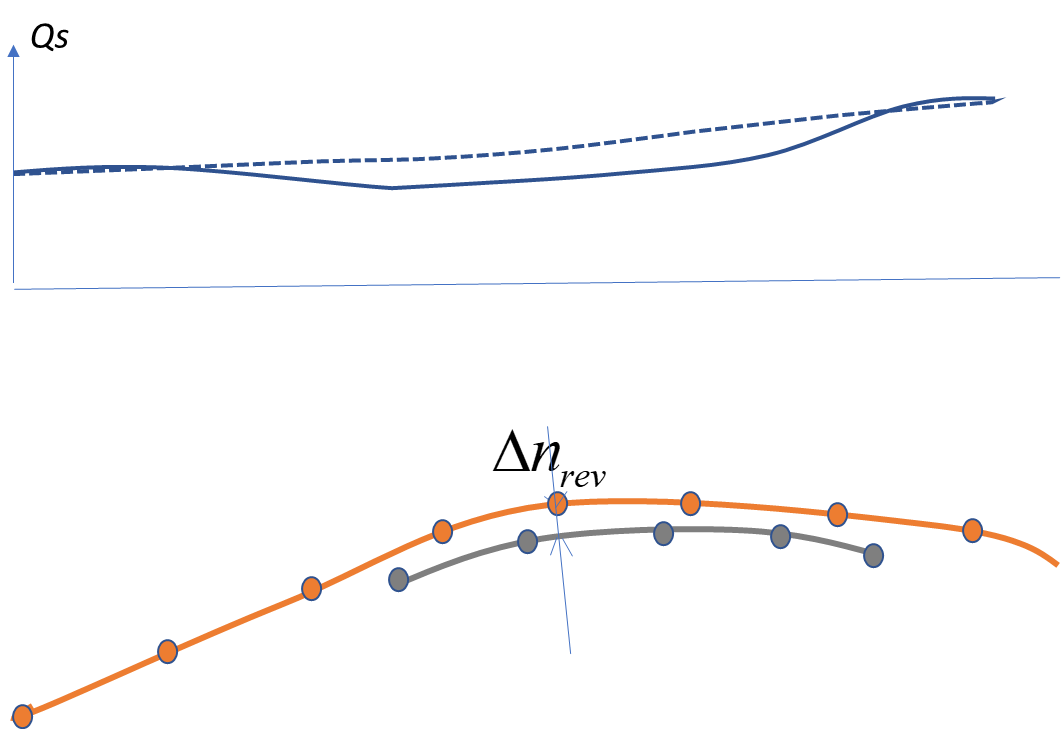


Figure 2 Definition of beach width (bottom panel) and example reduction of transport (top), where the dashed line is the original transport and the drawn line the modified transport rate.

# Input description

## Main script or text input file

The main function is called ShorelineS. This function is called with a state variable or structure S, which contains all necessary input for ShorelineS. In principle, there is no need to change anything in ShorelineS or lower functions, unless it is to fix bugs or add generic functionality.

One way to run ShorelineS is to create a master script that fills the state variable S with all the elements specific to a case. An example is the foowing script to run a circular island case:

|  |
| --- |
| addpath(genpath('..\functions\'))  %% MODEL INPUT PARAMETERS  S=struct;  S.reftime='2005-01-01'; % Reference time (i.e. 'yyyy-mm-dd')  S.endofsimulation='2009-01-01';  S.LDBcoastline='circle.txt';  S.WVCfile='wavedata\_IJMDMNTSPS\_200501010000\_201412312359.wvt';  S.d=6; % active profile height [m]  S.b=1e6; % CERC : coeff in simple cerc formula  S.dt=1/365/8; % time step in yr  S.tc=0; % Courant number automatic timestep; 0 for fixed step  S.ds0=200; % initial space step [m]  ….. *more input*  %% RUN SHORELINES MODEL  [S,O]=ShorelineS(S); |

An alternative way is to directly call runShorelineS(*inputfile)*, where the contents of *inputfile* would look like:

|  |
| --- |
| %% MODEL INPUT PARAMETERS  reftime='2005-01-01'; % Reference time (i.e. 'yyyy-mm-dd')  endofsimulation='2009-01-01';  LDBcoastline='circle.txt';  WVCfile='wavedata\_IJMDMNTSPS\_200501010000\_201412312359.wvt';  d=6; % active profile height [m]  b=1e6; % CERC : coeff in simple cerc formula  dt=1/365/8; % time step in yr  tc=0; % Courant number automatic timestep; 0 for fixed step  ds0=200; % initial space step [m]  ….. more input |

In the following we will use the second way of specifying the input, as it can also be applied with a compiled version of ShorelineS.

## Definition of coastline(s)

The coastline needs to be specified either as x-y vectors (in x\_mc and y\_mc as [1xN] vectors) or in a textfile with the xy-coordinates (in *LDBcoastline* with [Nx2] vector). Separate coastline segments can be specified, which need to be separated by NaN’s in the *x\_mc/y\_mc* or *LDBcoastline.* In all coastline sections, the land should be on the right when following the coastline point. Only cartesian coordinate systems can be used, with units in metres.

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.

Note that the prescribed elements are regridded in the model, which may increase or decrease the resolution depending on the specified grid cell size (*ds0*). In general, it is preferable to have a rather smooth initial coastline. Especially, if it is a coast with high-angle wave condition

## Model grid

A typical grid cell size (*ds0*) is in the order of 50-100. 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 (*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 *griddingmethod=1*. With this method it will also be easier to perform local grid refinement, which is being worked on.

A smoothing factor can be used which smooths the grid every timestep, but it is not recommended for most applications (by default 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: ds0=[x1,y1,ds1; x2,y2,ds2; ... ; xn,yn,dsn]

## Time frame

The start time has to be entered as yyyy-mm-dd (e.g. reftime='2020-01-01').

The end of the simulation is added as yyyy-mm-dd (e.g. endofsimulation='2040-01-01').

A variable timestep is used by default. It is automatically computed based on the transport rate You can set the fraction of this automatic timestep with *tc*. So, a *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 *tc=0* and setting *dt* to a value in years (e.g. *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 *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.

## Wave conditions

Wave conditions can either be specified as

* a static value,
* a time-series, or
* a wave climate with probability per condition

### Static condition

A static wave height can be specified in Hso (significant wave height in meters), the wave direction in phiw0 (as wave direction in nautical convention °N) and the wave period in tper (as peak wave period in seconds). A standard spreading around the mean-wave direction of 90 degrees is used (in spread) which can be adjusted.

### Single time series

A single time-series can be specified using  *WVCfile=’filename.wvt’*. The model checks the extension ‘wvt’ and then knows it is a time-serie 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  *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 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 probabilitie

Spatially varying time-series or wave climates can be specified by using a cell-string as input for the *WVCfile*. With the first column 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 file As an example, it may look like *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 or WVCfile containing a table with a reference to multiple WVT/WVC-files and their x and y locations as input for the *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 file As an example, it may look like *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-file

Alternatively, a NetCDF file can be used to provide the nearshore conditions, which is recognized from the ‘.nc’ extension in the *WVCfile*. It needs to contain variables 'station\_x', 'station\_y' for the location 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.

Looping the wave conditions is not yet possible.

The seeding number of randomized selection of wave climate conditions can be prescribed in *randomseed*, which allows for re-generation of the same random serie By default, this is off, meaning a different seeding number is taken for every new simulation.

## Foreshore orientation

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 wave

The foreshore orientation of the coastline can be specified using the *phif*. By default the foreshore orientation is not specified (*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. *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. *phif=[x1,y1,phif1; x2,y2,phif2; etc]* )

A option is also to use the current orientation of the coast but with some smoothing using a cell-string as input (e.g. *phif={'gaussian',7}* which smooths over 7 cells).

## Mapping of wave conditions to the shoreline

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:

* ‘interpolationmethod='alongshore\_mapping' which projects the wave locations on the coast and then interpolates alongshore, which is the most accurate but also the slowest.
* 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 *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 *ddeep* and the nearshore depth as *dnearshore*. Note that 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.

ddeep (at point of wave output) may in some cases be equal to 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*).

## Transport formulations

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 (*trform=’CERC’*) using a simple wave energy approach based on just the wave height (*Hso*) and wave direction (*phiw0*). *b* is a calibration factor for only this CERC formulation.
* CERC with implicit wave refraction (*trform=’CERC2’*) assuming a foreshore with same orientation as the coastline. This requires also the porosity (*por*), depth-induced breaking parameter (*gamma*) and density of the water and sediment (*rhow* and *rhos* in kg/m3).
* Kamphuis formulation (*trform=’KAMP’*) includes the wave period (*tper*), beach slope (*tanbeta*) and median grain size (*d50* in meters).
* Mil-Homens (*trform=’MILH’*)

Nearshore wave conditions at the point-of-breaking are used for:

* A modified nearshore CERC-formulation (*trform=’CERC3’*) which is similar to the regular CERC-formulation, but with the median grain size and adapted nearshore coefficient
* The Van Rijn 2015 formulation (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 (*Pswell*)

The wave transformation and transports are computed over the whole cross-shore profile for the Soulsby-VanRijn formulation with tide (*trform=’WAVETIDEPROF’*). This requires not along the specification of the waves at the boundary, but also the bed-friction coefficient (*Cf*), a minimum water depth (*hmin*) and closure depth for the tidal transport (*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 metre 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 (*tidefile*). The input is:

* x and y location of the stations (‘x\_stat’ and ‘y\_stat’), which are the first two column
* vertical amplitude of M2 and M4 tide components (‘eta’ in meter) in the 3rd 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 *qscal* parameter. A factor of 1 means no calibration of the transport rate

## Active profile height

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 extends 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 *(d)*. The active height is used in the code to translate the accumulation (or erosion) of sediment in m3/yr in a grid cell (with length ds) to a coastline change in m/yr.

A spatially varying active height can be specified (d=[x1,y1,hactive1; x2,y2,hactive2; etc])

It is noted that the active profile height and the transport rate (scaling) are co-varying.

## Coastline boundary conditions

You can specify open and closed boundaries for the start and end of the model in *boundary\_condition\_start* and *boundary\_condition\_end*.

The options for the boundaries are :

* closed boundary (=’closed’ or ={‘closed’,25000} for a 25 thousand m3/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')

Grid cells at the boundary will move perpendicular only to the initial orientation of the coastline.

## Nourishments

The nourishments can be specified in a ‘.nor’ file in *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 m3 (column 7)

Alternatively, a polygon may be defined with a polygon file (LDBnourish='example.pol' with nourishment locations [Nx2] and multiple nourishments separated by NaN’s). Furthermore, the rate of the nourishment (nourratefile='example\_rate.txt' in m3/yr), start and end date (nourstartfile='example\_start.txt'; 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 cell

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 described in the next section.

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.

## Shoreface nourishments

Shoreface norusihments slowly feed the coast and are therefore not applied instanteneously 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 (d50) [m]
* Placement moment tstart in [yyyymmdd]
* The total volume of the shoreface nourishment [m3]

## Revetments

The revetments can be specified in *x\_revet* and *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, *x\_revet=[x-values revetment 1 , NaN, x-values revetment 2, NaN, …]*;

Alternatively, a textfile with the revetments can be specified in *LDBrevetments* (as a string of the filename; i.e. instead of using the *x\_revet* and *y\_revet*). This textfile should contain a [Nx2] table of the xy-points of the revetments which is separated by NaN’s between the revetment.

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 *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 *crit\_width* is set to a small value of 5m 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 bypassing.

## Offshore breakwaters

An offshore breakwater (or groyne/breakwater without bypassing) can be specified in the *x\_hard* and *y\_hard*. This needs to be a series of x or y-points [1xN]. If you want to specify more offshore breakwaters, then please separate them with a NaN. So, *x\_hard=[x-values offshore breakwater 1 , NaN, x-values offshore breakwater 2, NaN, …]*.

Alternatively, a textfile with the hard structures can be specified in *LDBstructures* (as a string of the filename; i.e. instead of using the *x\_hard* and *y\_hard*). This textfile should contain a [Nx2] table of the xy-points of the offshore breakwaters which is separated by NaN’s between structures.

An offshore breakwater needs to cross the coast at most 1 time. If it crosses the coastline twice then the model will assume it is a groyne instead.

Wave diffraction can be switched on by using *diffraction=1*. The model plots will show tiny grey dots at coastline points with diffraction.

* 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).
* With *dirspr* the directional spreading can be specified, which adjust the type of diffraction. The model has been derived/tested only for small directional spreading (or swell waves) with a *dirspr* of 10 degree
* New derivations of wave diffraction parameters are currently made, which also include the effect of directional spreading on the reorientation of waves in the shadow zone (*Δωrot*) and the wave energy reduction (*kd*). Both were mentioned as being relevant in the meeting
* 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 circulation

Wave transmission at offshore breakwaters is under consideration.

Wave reflection at hard structures is not available yet.

## Groynes

A groyne with bypassing and diffraction needs to be specified in the *x\_hard* and *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, *x\_hard=[x-values groyne 1 , NaN, x-values groyne 2, NaN, …]*; Alternatively, a textfile with the hard structures can be specified in *LDBstructure*

The model will identify the structure as a groyne when it crosses the coast 2 time So, it is sort of a ‘staple’. Preferably, the first crossing is at a lower coastline index than the second. So, iot is first defined from land in seaward direction, then ‘to the right’ along the shore, and then back to land. If it crosses the coastline only once then the model will assume it is an offshore breakwater instead.

Wave diffraction can be switched on by using *diffraction=1*. The model plots will show tiny grey dots at coastline points with diffraction.

* With *dirspr* the directional spreading can be specified, which adjust the type of diffraction. The model has been derived/tested only for small directional spreading (or swell waves) with a *dirspr* of 10 degree
* 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 circulation

The bypassing of the groyne will depend on the ratio of the ‘point of breaking’ and the ‘depth at the breakwater-tip’.

* The point of breaking s computed by dividing the wave height (*Hs*) by the depth-induced breaking parameter (γ) and scaling it with the parameter *Aw*. By default, the model assumes an *Aw* of 5 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 *Aw* of 1 is needed 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 (*d50* in meters). The depth at breakwater tip = (1.04+0.086\*log(d50))^2 \* cross-shore extent^0.67.
* 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 *bypassdistribution\_power* with a higher value than 1. Two means a quadratic decay away from the groyne instead of linear.

Groynes are better not placed at the end of coastline section The coastline typically recedes inward over time, and therefore moves out of the groyne, resulting in a disconnect. So, defining the coastline just inside the end structure is risky. It is better to define the coast also at the other side of the groyne though. This aspect is still under consideration. In a new version we may force the coast at the boundary points to move perpendicular to the initial coastal orientation, which at least prevents the gradual recession towards the center out of the groyne.

It is planned to add a construction and decommissioning date for structures, to allow the model to place and remove structures over time.

A request was made to combine perpendicular and parallel structure This is noted, but has to be thought about well before being implemented.

## Dune interaction

Interaction between the coastline and the dunes is simulated following Larson et al. (2016), which considers the balance between onshore wind-driven transport and offshore dune erosion when the runup exceeds the dune foot elevation. To activate it, set *dune*=1. The initial beach width, dune foot elevation and dune crest elevation need to be specified alongshore, in a file marked by *LDBdune=<filename>.*  In this file, columns for x,y, beach width, dune foot elevation and dune crest elevation need to be specified. These properties are then interpolated alongshore in a way similar to nourishments or wave conditions. Output is generated for beach width *Wberm*, aeolian transport *qw* and dune erosionflux *qs.* The speed of dune erosion is regulated by a coefficient *Cs* with default value 7.5e-3.

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 *LDBdune* keyword it is possible to specify above 5 parameters separately (if *LDBdune=''*). Then the model will check for xdune, ydune, Wberm, Dfelev, Dcelev.

The following parameters (with default values) are relevant for dune erosion and dune growth:

kf=0.02 % Friction coefficient

Cs=5e-3 % Impact coefficient waves based on thesis M. Ghonim || very sensitive

d50r=2.5e-4 % Median reference grain size

rhoa=1.225 % Air density []

duneAw=0.1 % Coefficient (Bagnold, 1937)

Kw=4.2 % Empirical coefficient (Sherman et al. 2013)

k=0.41 % Von Karman's coefficient

segmaw=0.1 % fraction of the fetch used for the aeolian transport computation

Dune growth is driven by wind forcing. This requires either a wind time-series file or static wind forcing to be specified. The wind time-series can be specified in *WndCfile* and is an 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 (*uz)*, wind direction (*phiwind0*) and height at which wind is defined (*z*). Furthermore, a wind drag coefficient needs to be specified (*Cd*).

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 *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)

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 (*Watfile*) or as a fixed value (*WL0*). 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).

TODO: describe space- and time-varying surge and wave conditions for runup.

## Overwashing

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 (*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 (*Bheight*) and the closure-depth at the considered side (respectively *Dsf* at the seaward side and *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 *Dbb* is smaller than *Dsf*.

The rate of overwash is determined by a timescale factor (*OWtimescale* in years). The barrier width will exponentially develop towards the equilibrium width over the provided timescale/period.

TODO: describe overwash as part of Larson method

## Channel migration

TODO: describe channel migration mechanism

## High-angle instabilities

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 undulation

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 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 *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 *maxangle* can be adjusted to limit the angle change between grid cells. By default *maxangle=60°*.

With *plotUPW=1* it is possible to plot the locations where the high-angle correction is applied, which show up as green and red squares

There have been some issues with the high-angle waves in the model releases (for case in Sweden). These related to the foreshore orientation and some inconsistencies with structures and should, however, be resolved by now.

## Climate change

Sea level rise can be a constant rate (e.g. *ccSLR*=0.002 m/yr) or a table is used in *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 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. *ccHS*=0.001, which is +0.1% increase in HS per year). Or alternatively, a table is used in *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. *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 # degree

## Visualization of coastline and wave conditions

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 *plotHS* or *plotDIR* to show more detail of the wave height or wave direction at the coastline. *plotHS=6* will show you a value and arrow of the wave height at every 6th grid cell .While the *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 *plotQS* will show the transport rates along the coast.

The interval of plotting to screen can be set with *plotinterval* which contains the interval in number of timesteps (so, *plotinterval=2* means every two timesteps).

A time-interval can be set for exporting the figures (*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 *xlimits* and *ylimits* ([1x2] of min/max coordinate in m)

The position of the wave arrows and its scale can be set with *XYwave* ([1x2] coordinates).

An offset can be extracted from the x and y-coordinates in the plot with *XYoffset* [1x2]

The output dir of the plots and the data is placed in *outputdir* (e.g. 'Output\’)

The background fill of the coast can be switched on using *usefill* = 1 (or off when it is 0)

For complex coasts the *usefillpoints* can be set to a value larger than 0 to obtain a nice landfill behind the specified open coast.

fastplot=1 creates debug plots, which are made very quickly. While more detailed/slower plots can be made using fastplot=0.

In order to speed up the computation it may be decided to make the plot invisible by setting *plotvisible=0*. The plots will still be written to jpg-files in the output directory.

## Output data

An output file (‘output.mat’) will be exported by the model to the specified output directory (*outputdir*). The time-interval for the output mat-file (with O-structure) can be set in *storageinterval* which stores the interval in days. Note that it stores the data as actually used in the model, which means that the size of the coastline may change over time as a result of the adaptations of the grid. This will require the user to do some projection of the data on a grid.

The calculation of the cumulative transport can only be done when sufficient output timesteps are used. This has been brought forward as an issue for users. It is planned to allow the user to specify an output grid, on which results are projected. This would also allow for a cumulative Q This is likely a priority for a future version.

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.