Keywords

Place keywords here

Summary

Place summary here

References

Place references here

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Version | Date | Author | Initials | Review | Initials | Approval | Initials |
|  | jan. 2015 |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |

|  |
| --- |
| State  draft  This is a draft report, intended for discussion purposes only. No part of this report may be relied upon by either principals or third parties. |

Contents

1 Introduction 1

2 Processes and model formulation 2

2.1 Domain and definitions 2

2.2 Hydrodynamics options 2

2.2.1 Stationary mode 2

2.2.2 Non-stationary (surfbeat) mode 2

2.2.3 Wave resolving mode 2

2.3 Short wave propagation 3

2.3.1 Wave action balance 3

2.3.2 Dissipation 5

2.3.3 Roller energy balance 8

2.4 Shallow water equations 8

2.5 Nonhydrostatic pressure correction 8

2.6 Groundwater flow 8

2.7 Bedload transport 8

2.8 Suspended load transport 8

2.9 Bottom updating 8

2.9.1 Due to sediment fluxes 8

2.9.2 Avalanching 8

2.9.3 Bed composition 8

3 Numerical implementation 9

3.1 Grid types 9

3.1.1 1D 9

3.1.2 Rectilinear 9

3.1.3 Curvilinear 9

3.2 Wave action balance 9

3.2.1 Stationary solver 9

3.2.2 Nonstationary solver 9

3.3 Shallow water equations 9

3.4 Nonhydrostatic pressure correction 9

3.5 Advection-diffusion equation for sediment 9

3.6 Bottom updating schemes 9

3.7 Avalanching 9

3.8 Bed composition 9

4 Boundary conditions 10

4.1 Waves 10

4.1.1 Time series 10

4.1.2 Spectra 10

4.1.3 Lateral boundary conditions 10

4.2 Shallow water equations 10

4.2.1 Absorbing-generating 10

4.2.2 River and point discharge 10

4.2.3 Ship motion 10

4.2.4 Lateral boundaries 10

4.2.5 Tide and surge 10

4.3 Sediment transport 10

5 Input description 11

5.1 General 11

5.2 Physical processes 11

5.3 Grid and bathymetry 12

5.4 Waves input 15

5.4.1 Stationary wave boundary conditions 18

5.4.2 Spectral wave boundary conditions 19

5.4.3 Boundary conditions for non-hydrostatic model 26

5.4.4 Special types of wave boundary conditions 26

5.4.5 Temporally and/or spatially varying wave boundary conditions 26

5.4.6 Notes on the generation of wave boundary conditions 28

5.5 Tide and surge input 28

5.5.1 Tide boundary conditions 28

5.5.2 Flow boundary conditions 29

5.6 Water level (dam break) 30

5.7 Wind input 30

5.8 Sediment input 31

5.9 Vegetation input 32

5.10 Discharge input 32

5.11 Drifters input 33

5.12 Ship input 33

5.13 Output selection 33

5.14 Time parameters 36

5.15 Model coefficients 36

5.15.1 Wave numerics 36

5.15.2 Wave dissipation 37

5.15.3 Rollers 38

5.15.4 Wave-current interaction 38

5.15.5 Flow 38

5.15.6 Flow numerics 39

5.15.7 Sediment transport 40

5.15.8 Sediment transport numerics 44

5.15.9 Quasi-3D sediment transport 45

5.15.10 Morphology 45

5.15.11 Bed update 46

5.15.12 Groundwater flow 47

5.15.13 Non-hydrostatic correction 48

5.15.14 Physical constants 50

5.15.15 Coriolis force 50

5.15.16 MPI 50

5.15.17 Output projection 51

6 Bibliography 52

7 Tutorial 53

7.1 1-D profile model 53

7.2 2-D area model 53

7.3 Langsgetij + riveroutflow 53

# Introduction

Dano, Ad, Ap

# Processes and model formulation

## Domain and definitions

Dano - overnemen en nieuw plaatje curvi

## Hydrodynamics options

Dano

### Stationary mode

### Non-stationary (surfbeat) mode

### Wave resolving mode

## Short wave propagation

### Wave action balance

Kees - bezig

The wave forcing in the shallow water momentum equation is obtained from a time dependent version of the wave action balance equation. Similar to Delft University’s (stationary) HISWA model (Holthuijsen et al., 1989) the directional distribution of the action density is taken into account whereas the frequency spectrum is represented by a frequency, best represented by the spectral parameter *fm-1,0*.The wave action balance is then given by:



In which the wave action *A* is calculated as:



In *θ* represents the angle of incidence with respect to the x-axis, *Sw* represents the wave energy density in each directional bin and *σ* the intrinsic wave frequency. The wave action propagation speeds in x- and y-direction are given by:



With *uL* and *vL* the cross-shore and alongshore depth-averaged Lagrangian velocities respectively (defined below), and the group velocity cg obtained from linear theory. If wave-current interaction is turned off (*wci=0*) then the last term in either equation is not taken into account. The propagation speed in θ-space is obtained from:



In *h* represents the total water depth and in this formulation bottom refraction (first term) and wave-current interaction (last two terms) are taken into account. If wave-current interaction is turned off (*wci=0*) then the last two terms are neglected.

The wave number *k* is obtained from the eikonal equations that is described in . In this formulation the subscripts refer to the direction of the wave vector components and *ω* represents the absolute radial frequency.



The wave number is then obtained from .



The absolute radial frequency *ω* is given by . The intrinsic frequency *σ* is obtained from the linear dispersion relation. If wave-current interaction is turned off (*wci=0*) then the last two terms are not taken into account.



### Dissipation

#### Breaking

Kees - bezig

There are in four different wave breaking formulations implemented in XBeach. The formulations are coded with the keyword *break*.

1. Non-stationary waves: formulation of Roelvink (1993a)
2. Stationary waves: formulation of Baldock et al. (1998)
3. Non-stationary waves: adaptation of break=1
4. Non-stationary waves: adaptation of break=1 (Daly et al. ,2010)

For the non-stationary (surf beat) approach the total wave energy dissipation, i.e. directionally integrated, due to wave breaking is modelled according to Roelvink (1993a). This is coded as *break=1*. In *α* is applied as wave dissipation coefficient, *Qb* is the fraction breaking waves, *p* stands for the water density and *γ* is the breaker index. The total wave energy *Ew* is calculated by integrating over the wave direction per directional bin.



In a variation of , one could also use the third wave breaking formulation, presented in . This formulation is somewhat different than the formulation of Roelvink (1993a). This is coded as *break=3.*



On top of that, Daly et al. (2010) developed a formulation presented in , which states that waves are fully breaking if the wave height exceeds a threshold (*γ*) and stop breaking if the wave height fall below another threshold (*γ2*). This is coded as *break=4*.



In the stationary case Baldock et al. (1998) is applied, which is presented in . In this breaking formulation the fraction breaking waves *Qb* and breaking wave height *Hb* is calculated differently compared to the breaking formulations used for the non-stationary situation. In *α* is applied as wave dissipation coefficient, *frep* represents a representative intrinsic frequency and *y* is a calibration factor. The stationary wave breaking formulation is coded with *break=4*.



In either the non-stationary or stationary case the total wave dissipation is distributed proportionally over the wave directions with the formulation in .



#### Bottom friction

Kees

#### Vegetation

Arnold

### Roller energy balance

Dano

## Shallow water equations

Kees

## Nonhydrostatic pressure correction

Robert

## Groundwater flow

Kees/Robert

## Bedload transport

Kees + Lodewijk

## Suspended load transport

Kees + Lodewijk

## Bottom updating

### Due to sediment fluxes

Kees

### Avalanching

Kees + Pieter

### Bed composition

Bas

# Numerical implementation

Dano behalve 3.4,3.8

## Grid types

### 1D

### Rectilinear

### Curvilinear

## Wave action balance

### Stationary solver

### Nonstationary solver

## Shallow water equations

## Nonhydrostatic pressure correction

Robert

## Advection-diffusion equation for sediment

## Bottom updating schemes

## Avalanching

## Bed composition

Bas

# Boundary conditions

## Waves

### Time series

Kees, Ap review

### Spectra

Kees, Ap review

### Lateral boundary conditions

Dano

## Shallow water equations

### Absorbing-generating

Ap met appendix

### River and point discharge

Bas

### Ship motion

Dano

### Lateral boundaries

Kees

### Tide and surge

Kees

## Sediment transport

Dano

# Input description

Bas - params en attribute files

## General

Upon execution of the XBeach executable *xbeach.exe* the file *params.txt* in the current working directory will be read. The *params.txt* file contains keyword/value pairs that determine the parameter settings of XBeach. Each keyword/value pair may contain an actual model parameter or refers to another file with additional information on the model setup. If a *params.txt* file cannot be found then the execution of XBeach will be aborted.

In the *params.txt* file there can be a single keyword/value pair per line in any order. A keyword/value pair is separated by an equal sign (=). Each line containing an equal sign is interpreted as a keyword/value pair. Reversely, any lines without an equal sign are ignored and may be used for comments. Only a few keywords are required, others have default values that are used in case the keyword is not mentioned in the *params.txt* file.

The *params.txt* file contains grid and bathymetry info, wave input, flow input, morphological input, et cetera. This chapter describes the possibilities of the *params.*txt file and any files that can be referred to from the *params.txt* file. The tables in this chapter contain a description of the keywords, the default values, its units and recommended value ranges, while the formats for additional input files are described in the relevant sections. Keywords marked with an asterix (\*) are considered advanced options and should not be used for regular applications of XBeach.

## Physical processes

XBeach supports a variety of physical processes from generic, like waves and flow, to very specific, like ship motions and point discharge. Each process can be switched on or off. The commonly used processes are turned on by default. The table below lists the keywords used to switch on or off physical processes in XBeach.

| keyword | description | default | range | units | remark |
| --- | --- | --- | --- | --- | --- |
| avalanching | Turn on avalanching | 1 | 0 - 1 | - |  |
| bchwiz | Turn on beachwizard | 0 | 0 - 1 | - |  |
| flow | Turn on flow calculation | 1 | 0 - 1 | - |  |
| gwflow\* | Turn on groundwater flow | 0 | 0 - 1 | - |  |
| lwave | Turn on short wave forcing on NLSW equations and boundary conditions | 1 | 0 - 1 | - |  |
| morphology | Turn on morphology | 1 | 0 - 1 | - |  |
| nonh\* | Turn on non-hydrostatic pressure: 0 = NSWE, 1 = NSW + non-hydrostatic pressure compensation Stelling & Zijlema, 2003 | 0 | 0 - 1 | - |  |
| q3d\* | Turn on quasi-3D sediment transport | 0 | 0 - 1 | - |  |
| sedtrans | Turn on sediment transport | 1 | 0 - 1 | - |  |
| setbathy | Turn on timeseries of prescribed bathy input | 0 | 0 - 1 | - |  |
| ships\* | Turn on ship waves | 0 | 0 - 1 | - |  |
| single\_dir\* | Turn on stationary model for refraction, surfbeat based on mean direction | 0 | 0 - 1 | - |  |
| snells\* | Turn on Snell's law for wave refraction | 0 | 0 - 1 | - |  |
| swave | Turn on short waves | 1 | 0 - 1 | - |  |
| swrunup\* | Turn on short wave runup | 0 | 0 - 1 | - |  |
| vegetation\* | Turn on interaction of waves and flow with vegetation | 0 | 0 - 1 | - |  |

## Grid and bathymetry

XBeach’ spatial grid size is defined by the keywords *nx* and *ny*. The size of the computational grid will be *nx+1* by *ny+1* cells large. The initial bathymetry is provided using a separate file that is referred to by the *depfile* keyword. This file contains an initial bed level for each grid cell where each line corresponds to a transect in x-direction (cross-shore). The values are positive down by default, but this can be changed using the *posdwn* keyword.

Three main types of XBeach grids are supported: fast 1D, 1D and 2DH. Fast 1D grids have a single alongshore grid cell and thus a value *ny=0* and a single line in the *depfile*. The 1D grids have 3 alongshore grid cells and thus a value *ny=2* and three lines in the *depfile*. The 2DH grids have more than 3 alongshore grid cells, a value *ny>2* and as may lines in the *depfile*.

XBeach spatial grids can be equidistant or non-equidistant. In the former case the grid size is defined by the keywords *dx* and *dy*. In the latter case the keyword *vardx* should be set to *1* and x- and y-coordinates of the grid cells should be provided through the files referenced by the *xfile* and *yfile* keywords. These files take exactly the same format as the *depfile* file where all coordinates along the x-direction are on one line and each line represents a cell in y-direction. XBeach grids are defined in a coordinate system of choice and can be either rectangular or curvilinear grids.

Delft3D grids created with tools like RFGRID are also supported. To use Delft3D grids, choose *gridform=delft3d* and provide a grid file via the keyword *xyfile*. The format of Delft3D grids is not described here, but can be found in the Delft3D manual. Also forced updating of bathymetries is supported as described in section 5.15.11 Bed update.

Apart for the spatial grid, XBeach also uses a directional grid for short waves and rollers. The grid is determined by a minimum and maximum angle and a directional bin size using the keywords *thetamin*, *thetamax* and *dtheta* respectively. The *thetamin* and *thetamax* angles are either defined according to the Cartesian convention (angle w.r.t. the computational x-axis) or according to the nautical convention (angle w.r.t. deg. N, so from W is 270 deg. N). The convention is chosen using the keyword *thetanaut*.

Examples of typical input for a non-equidistant, fast 1D XBeach model are:

params.txt

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%%% XBeach parameter settings input file %%%

%%% %%%

%%% date: 01-Jan-2015 12:00 %%%

%%% function: xb\_write\_params %%%

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%%% Grid parameters %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

depfile = bed.dep

posdwn = 0

nx = 265

ny = 0

alfa = 0

vardx = 1

xfile = x.grd

yfile = y.grd

thetamin = -90

thetamax = 90

dtheta = 15

thetanaut = 0

bed.dep

1. 0.09 0.46 0.80 1.14 1.46 1.77 ... 29.14 29.12 29.10 29.07 29.06 29.05

x.grd

0.00 17.38 34.77 52.15 69.54 ... 1403.32 1407.88 1412.44 1417.00 1421.56

y.grd

0.00 0.00 0.00 0.00 0.00 0.00 0.00 ... 0.00 0.00 0.00 0.00 0.00 0.00 0.00

All keywords related to grid and bathymetry input are listed in the following table:

| keyword | description | default | range | units | remark |
| --- | --- | --- | --- | --- | --- |
| alfa | Angle of x-axis from East | 0.0 | 0.0 - 360.0 | deg |  |
| depfile | Name of the input bathymetry file |  |  | <file> |  |
| dtheta | Directional resolution | 10.0 | 0.1 - 20.0 | deg |  |
| dtheta\_s | Directional resolution in case of stationary refraction | 10.0 | 0.1 - 20.0 | deg |  |
| dx | Regular grid spacing in x-direction | -1.0 | 0.0 - 1000000000.0 | m |  |
| dy | Regular grid spacing in y-direction | -1.0 | 0.0 - 1000000000.0 | m |  |
| gridform | Grid definition format | xbeach | xbeach, delft3d |  |  |
| nx | Number of computational cell corners in x-direction | 50 | 2 - 10000 | - |  |
| ny | Number of computational cell corners in y-direction | 2 | 0 - 10000 | - |  |
| posdwn | Bathymetry is specified positive down (1) or positive up (-1) | 1.0 | -1.0 - 1.0 | - |  |
| thetamax | Higher directional limit (angle w.r.t computational x-axis) | 90.0 | -180.0 - 180.0 | deg |  |
| thetamin | Lower directional limit (angle w.r.t computational x-axis) | -90.0 | -180.0 - 180.0 | deg |  |
| thetanaut | Switch to specify thetamin and thetamax in nautical convention rather than cartesian | 0 | 0 - 1 | - |  |
| vardx | Switch for variable grid spacing | 0 | 0 - 1 | - |  |
| xfile | Name of the file containing x-coordinates of the calculation grid |  |  | <file> |  |
| xori | X-coordinate of origin of axis | 0.0 | -100000000.0 - 1000000000.0 | m |  |
| xyfile | Name of the file containing Delft3D xy-coordinates of the calculation grid |  |  | <file> |  |
| yfile | Name of the file containing y-coordinates of the calculation grid |  |  | <file> |  |
| yori | Y-coordinate of origin of axis | 0.0 | -1000000000.0 - 1000000000.0 | m |  |

## Waves input

An XBeach model is generally forced by waves on its offshore boundary. These waves are described by the wave boundary conditions discussed in this section. The details of the wave motions within the model are described by the wave numerics in terms of the wave action balance (see 5.15.1 Wave numerics) and wave-current interaction (5.15.4 Wave-current interaction), wave dissipation model (see 5.15.2 Wave dissipation) and wave roller model (5.15.3 Rollers).

XBeach supports a variety of wave boundary condition types that are divided in two main groups: stationary and spectral boundary conditions. The *instat* keyword can be used to select one particular type of wave boundary conditions. Table XXX gives an overview of all types of wave boundary conditions available for XBeach. Figure XXX can be used to help you determine what type of wave boundary conditions is appropriate for your case. Each wave boundary condition type is explained in the following subsections. Note that most spectral wave boundary conditions can vary both in space and time using a *FILELIST* and/or *LOCLIST* construction as described in 5.4.4 Temporally and/or spatially varying wave boundary conditions.

Table XXX Overview of wave boundary conditions supported by XBeach

|  |  |
| --- | --- |
| *instat* | description |
| off | no wave boundary condition |
| stat | stationary wave boundary condition (sea state) |
| bichrom | bichromatic (two wave component) waves |
| ts\_1 | first-order timeseries of waves (generated outside XBeach) |
| ts\_2 | second-order timeseries of waves (generated outside XBeach) |
| jons | wave groups generated using a parametric (Jonswap) spectrum |
| swan | wave groups generated using a SWAN 2D output file |
| vardens | wave groups generated using a formatted file |
| nonh | boundary conditions for nonhydrostatic option |
| reuse | reuse of wave conditions |
| stat\_table | a sequence of stationary conditions (sea states) |
| jons\_table | a sequence of time-varying wave groups |

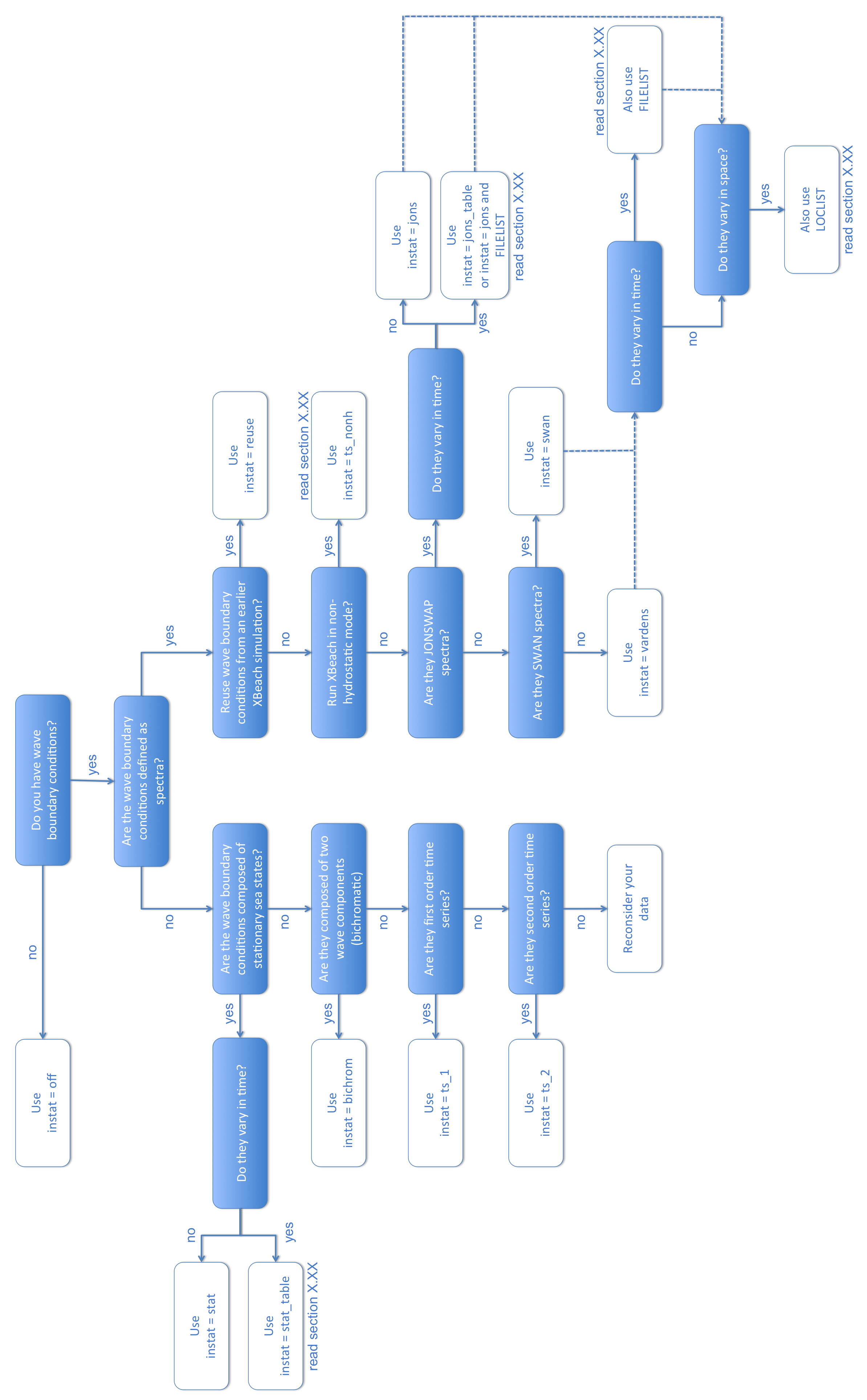


Figure XXX Decision tree for selecting the appropriate type of wave boundary conditions

### Stationary wave boundary conditions

Stationary wave boundary conditions are enabled using *instat* values *stat*, *bichrom*, *ts\_1*, *ts\_2* or *stat\_table*. The conditions aregenerally defined within the *params.txt* file directly using the keywords described in the table below. In addition, in case of *instat* values *ts\_1* or *ts\_2* the file *bc/gen.ezs* should be present that describes the infragravity wave forcing generated outside of XBeach. The format of this file is described in 5.5.2 Flow boundary conditions.

The only exception is the case of *instat=stat\_table* where time-varying stationary wave boundary conditions are described in an external file referenced by the *bcfile* keyword. The *bcfile* keyword is part of the spectral wave boundary condition input and also the referenced file is designed for time-varying spectral input in the form of JONSWAP spectra. In stationary mode only the relevant data from this file is used and irrelevant data like *gamma* and *dfj* are discarded. See 5.4.2.1 JONSWAP wave spectra for a description of the file format.

| keyword | description | default | range | units | remark |
| --- | --- | --- | --- | --- | --- |
| Hrms | Hrms wave height for instat = stat, bichrom, ts\_1 or ts\_2 | 1.0 | 0.0 - 10.0 | m |  |
| Tlong | Wave group period for case instat = bichrom | 80.0 | 20.0 - 300.0 | s |  |
| Trep | Representative wave period for instat = stat, bichrom, ts\_1 or ts\_2 | 10.0 | 1.0 - 20.0 | s |  |
| dir0 | Mean wave direction for instat = stat, bichrom, ts\_1 or ts\_2 (nautical convention) | 270.0 | 180.0 - 360.0 | deg |  |
| instat | Wave boundary condition type | bichrom | stat, bichrom, ts\_1, ts\_2, jons, swan, vardens, reuse, ts\_nonh, off, stat\_table, jons\_table |  |  |
| lateralwave | Switch for lateral boundary at left | neumann | neumann, wavecrest, cyclic |  |  |
| m | Power in cos^m directional distribution for instat = stat, bichrom, ts\_1 or ts\_2 | 10 | 2 - 128 | - |  |
| nmax\* | Maximum ratio of cg/c for computing long wave boundary conditions | 0.8 | 0.5 - 1.0 | - |  |
| taper | Spin-up time of wave boundary conditions, in morphological time | 100.0 | 0.0 - 1000.0 | s |  |

### Spectral wave boundary conditions

Spectral wave boundary conditions are enabled using *instat* values *jons*, *swan*, *vardens* or *jons\_table*. The conditions are defined in separate files referenced from the *params.txt* file using the *bcfile* keyword. A spectral wave boundary condition describes a spectrum *shape* that XBeach uses to generate a (random) wave time series. The length and resolution of the generated time series is determined by the keywords *rt* and *dtbc* respectively. XBeach will reuse the generated time series until the simulation is completed. The resolution of the time series should be enough to accurately represent the bound long wave, but need not be as small as the time step used in XBeach.

An overview of all keywords relevant for spectral wave boundary conditions is given in the table below. The necessary file formats for each type of spectral wave boundary condition is explained in the following subsections.

| keyword | description | default | range | units | remark |
| --- | --- | --- | --- | --- | --- |
| Tm01switch\* | Switch to enable Tm01 rather than Tm-10 | 0 | 0 - 1 | - |  |
| bcfile | Name of spectrum file |  |  | <file> |  |
| correctHm0\* | Switch to enable Hm0 correction | 1 | 0 - 1 | - |  |
| dtbc\* | Timestep used to describe time series of wave energy and long wave flux at offshore boundary (not affected by morfac) | 1.0 | 0.1 - 2.0 | s |  |
| dthetaS\_XB\* | The (counter-clockwise) angle in the degrees needed to rotate from the x-axis in SWAN to the x-axis pointing East | 0.0 | -360.0 - 360.0 | deg |  |
| fcutoff\* | Low-freq cutoff frequency for instat = jons, swan or vardens boundary conditions | 0.0 | 0.0 - 40.0 | Hz |  |
| instat | Wave boundary condition type | bichrom | stat, bichrom, ts\_1, ts\_2, jons, swan, vardens, reuse, ts\_nonh, off, stat\_table, jons\_table |  |  |
| nonhspectrum\* | Spectrum format for wave action balance of nonhydrostatic waves | 0 | 0 - 1 | - |  |
| nspectrumloc\* | Number of input spectrum locations | 1 | 1 - par%ny+1 | - |  |
| nspr\* | Switch to enable long wave direction forced into centres of short wave bins | 0 | 0 - 1 | - |  |
| oldnyq\* | Switch to enable old nyquist switch | 0 | 0 - 1 | - |  |
| random\* | Switch to enable random seed for instat = jons, swan or vardens boundary conditions | 1 | 0 - 1 | - |  |
| rt | Duration of wave spectrum at offshore boundary, in morphological time | min(3600.d0 | par%tstop - None | s |  |
| sprdthr\* | Threshold ratio to maximum value of S above which spectrum densities are read in | 0.08 | 0.0 - 1.0 | - |  |
| trepfac\* | Compute mean wave period over energy band: par%trepfac\*maxval(Sf) for instat jons, swan or vardens; converges to Tm01 for trepfac = 0.0 and | 0.01 | 0.0 - 1.0 | - |  |
| wbcversion\* | Version of wave boundary conditions | 3 | 1 - 3 | - |  |

#### JONSWAP wave spectra

JONSWAP spectrum input is enabled using *instat=jons*. A JONSWAP wave spectrum is parametrically defined in a file that is referenced using the *bcfile* keyword. This file contains a single parameter per line in arbitrary order. The parameters that can be defined are listed in Table XXX. All variables are optional. If no value is given, the default value as specified in the table is used. It is advised not to specify the keyword *dfj* and allow XBeach to calculate the default value.

A typical JONSWAP definition file looks as follows:

jonswap.txt

Hm0 = 0.8

fp = 0.125

mainang = 285.

gammajsp = 3.3

s = 10.

fnyq = 0.3

It is possible to use an alternative file format for time-varying JONSWAP spectra. To enable this option use the *instat* value *jons\_table*. In this case, each line in the spectrum definition file contains a parametric definition of a spectrum, like in a regular JONSWAP definition file, plus the duration for which that spectrum is used during the simulation. XBeach does not reuse time-varying spectrum files. Therefore the total duration of all spectra should at least match the duration of the simulation. The name of the file can be chosen freely, but the file format is fixed as follows and all parameters should be present in all lines:

jonswap.txt

<Hm0> <Tp> <mainang> <gammajsp> <s> <duration> <dtbc>

Note that we refer to the keywords used in a regular JONSWAP definition file in this example, with three differences: 1) the peak period rather than the peak frequency is defined 2) the duration is added (similar to *rt* in *params.txt*) 3) the time resolution is added (similar to *dtbc* in *params.txt*). The duration and boundary condition time step in this file overrules *rt* and *dtbf* in *params.txt*. This format is also used for time-varying stationary wave boundary conditions as described in 5.4.1 Stationary wave boundary conditions. As an example, the JONSWAP spectrum definition file presented above would look as follows if the significant wave height should be increased with 0.2 m every hour:

jonswap.txt

0.8 8. 285. 3.3 10. 0.3 3600. 0.05

1.0 8. 285. 3.3 10. 0.3 3600. 0.05

1.2 8. 285. 3.3 10. 0.3 3600. 0.05

A more generic way of providing time-varying spectral wave boundary conditions is using a FILELIST construction as described in 5.4.4 Temporally and/or spatially varying wave boundary conditions. This approach is compatible with all spectral wave boundary condition types as well as spatially varying boundary conditions as described in the same section.

Table XXX Overview of available keywords in JONSWAP definition file

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| keyword | description | default | minimum | maximum |
| Hm0 | Hm0 of the wave spectrum, significant wave height [m] | 0.0 | 0.0 | 5.0 |
| fp | Peak frequency of the wave spectrum [s-1] | 0.08 | 0.0625 | 0.4 |
| gammajsp | Peak enhancement factor in the JONSWAP expression [-] | 3.3 | 1.0 | 5.0 |
| s | Directional spreading coefficient, cosine law [-] | 10. | 1.0 | 1000. |
| mainang | Main wave angle (nautical convention) [°] | 270. | 180. | 360. |
| fnyq | Highest frequency used to create JONSWAP spectrum [s-1] | 0.3 | 0.2 | 1.0 |
| dfj | Step size frequency used to create JONSWAP spectrum [s-1] | fnyq/200 | fnyq/1000 | fnyq/20 |

#### SWAN wave spectra

XBeach can read standard SWAN 2D variance density or energy density output files (\*.sp2 files) as specified in the SWAN v40.51 manual. This option is enabled using *instat=swan* in *params.txt* and a reference to the spectrum file via the keyword *bcfile*. XBeach assumes the directional information in the SWAN file is according to the nautical convention. If the file uses the Cartesian convention for directions, the user must specify the angle in degrees to rotate the x-axis in SWAN to the x-axis in XBeach (by the Cartesian convention). This value is specified in *params.txt* using the keyword *dthetaS\_XB*.

Note that time-varying and spatially varying SWAN spectra can be provided using the FILELIST and LOCLIST constructions as described in 5.4.4 Temporally and/or spatially varying wave boundary conditions.

An example of a 2D SWAN spectrum is given below:

swan.txt

SWAN 1 Swan standard spectral file

$ Data produced by SWAN version 40.51

$ Project:'projname' ; run number:'runnum'

LOCATIONS locations in x-y-space

1 number of locations

22222.22 0.00

RFREQ relative frequencies in Hz

23 number of frequencies

0.0545

0.0622

0.0710

0.0810

0.0924

0.1055

0.1204

0.1375

0.1569

0.1791

0.2045

0.2334

0.2664

0.3040

0.3470

0.3961

0.4522

0.5161

0.5891

0.6724

0.7675

0.8761

1.0000

CDIR spectral Cartesian directions in degr

12 number of directions

30.0000

60.0000

90.0000

120.0000

150.0000

180.0000

210.0000

240.0000

270.0000

300.0000

330.0000

360.0000

QUANT

1 number of quantities in table

VaDens variance densities in m2/Hz/degr

m2/Hz/degr unit

-0.9900E+02 exception value

FACTOR

0.675611E-06

51 242 574 956 1288 1482 1481 1286 957 579 244 51

129 610 1443 2402 3238 3725 3724 3234 2406 1454 613 128

273 1287 3054 5084 6846 7872 7869 6837 5091 3076 1295 271

665 3152 7463 12402 16712 19229 19221 16690 12419 7518 3172 662

1302 6159 14608 24275 32688 37618 37603 32644 24309 14716 6198 1296

2328 10989 26020 43341 58358 67109 67080 58281 43401 26213 11058 2317

3365 15922 37712 62733 84492 97150 97110 84380 62820 37991 16021 3349

3426 16230 38440 63939 86109 99010 98969 85995 64027 38724 16331 3410

2027 9612 22730 37790 50909 58529 58505 50841 37843 22898 9672 2018

672 3178 7538 12535 16892 19440 19432 16870 12552 7594 3198 669

101 479 1135 1890 2542 2924 2923 2539 1892 1144 482 101

2 11 26 43 57 66 66 57 43 26 11 2

0 0 0 0 0 0 0 0 0 0 0 0

0 0 0 0 0 0 0 0 0 0 0 0

0 0 0 0 0 0 0 0 0 0 0 0

0 0 0 0 0 0 0 0 0 0 0 0

0 0 0 0 0 0 0 0 0 0 0 0

0 0 0 0 0 0 0 0 0 0 0 0

0 0 0 0 0 0 0 0 0 0 0 0

0 0 0 0 0 0 0 0 0 0 0 0

0 0 0 0 0 0 0 0 0 0 0 0

0 0 0 0 0 0 0 0 0 0 0 0

0 0 0 0 0 0 0 0 0 0 0 0

0 0 0 0 0 0 0 0 0 0 0 0

#### Variance density spectra

2D spectral information that is not in SWAN format can be provided using a formatted variance density spectrum file and *instat=vardens*. The spectrum file itself is again referenced using the keyword *bcfile*. The contents of the file must adhere to a specific format:

vardens.txt

<number of frequencies (n)>

<frequency 1>

<frequency 2>

<frequency 3>

...

<frequency n-1>

<frequency n>

<number of directions (m)>

<directions 1>

<directions 2>

<directions 3>

...

<directions m-1>

<directions m>

<variance density 1,1> <variance density 2,1> ... <variance density m,1>

<variance density 1,2> <variance density 2,2> ... <variance density m,2>

...

<variance density 1,n> <variance density 2,n> ... <variance density m,n>

Note that the directions must defined according to the Cartesion convention and in the coordinate system used by XBeach. In this coordinate system 0° corresponds to the direction of the x-axis, while 90° corresponds to the direction of the y-axis. Also, the directions must be defined in increasing order. Time-varying and spatially varying variance density spectra can be provided using the FILELIST and LOCLIST constructions as described in 5.4.4 Temporally and/or spatially varying wave boundary conditions.

An example of a formatted variance density file is given below:

vardens.txt

15

0.0418

0.0477

0.0545

0.0622

0.0710

0.0810

0.0924

0.1055

0.1204

0.1375

0.1569

0.1791

0.2045

0.2334

0.2664

13

-180.0000

-150.0000

-120.0000

-90.0000

-60.0000

-30.0000

0.0000

30.0000

60.0000

90.0000

120.0000

150.0000

180.0000

0 0 0 0 0 0 0 0 0 0 0 0

51 242 574 956 1288 1482 1481 1286 957 579 244 51

129 610 1443 2402 3238 3725 3724 3234 2406 1454 613 128

273 1287 3054 5084 6846 7872 7869 6837 5091 3076 1295 271

665 3152 7463 12402 16712 19229 19221 16690 12419 7518 3172 662

1302 6159 14608 24275 32688 37618 37603 32644 24309 14716 6198 1296

2328 10989 26020 43341 58358 67109 67080 58281 43401 26213 11058 2317

3365 15922 37712 62733 84492 97150 97110 84380 62820 37991 16021 3349

3426 16230 38440 63939 86109 99010 98969 85995 64027 38724 16331 3410

2027 9612 22730 37790 50909 58529 58505 50841 37843 22898 9672 2018

672 3178 7538 12535 16892 19440 19432 16870 12552 7594 3198 669

101 479 1135 1890 2542 2924 2923 2539 1892 1144 482 101

2 11 26 43 57 66 66 57 43 26 11 2

0 0 0 1 1 1 1 1 1 0 0 0

0 0 0 0 0 0 0 0 0 0 0 0

0 0 0 0 0 0 0 0 0 0 0 0

### Boundary conditions for non-hydrostatic model

If XBeach is run as a non-hydrostatic model, which is essentially the nonlinear shallow water equations with dispersion terms and without a wave-action driver, appropriate wave boundary conditions need to be supplied. This option is enabled by *instat=nonh*.

TODO: nonh wbc description

### Special types of wave boundary conditions

Two special types of wave boundary conditions are available that makes XBeach skip the generation of new wave time series. The first is *instat=off* which simply does not provide any wave forcing on the model and hence no wave action in the model.

The second is *instat=reuse* which makes XBeach reuse wave time series that were generated during a previous simulation. This can be a simulation using the same or a different model as long as the computational grids are identical. In order to reuse boundary conditions, all relevant files should be copied to the current working directory of the model (where the *params.txt* file is located). Relevant files are the *ebcflist.bcf* and *qbcflist.bcf* files and all files referenced therein. Generally, the referenced files have *E\_* and *q\_* prefixes. No further wave boundary condition data need be given in *params.txt*.

### Temporally and/or spatially varying wave boundary conditions

Time-varying spectral wave boundary conditions can be defined by feeding in multiple spectrum definition files rather than a single definition file. In addition, the duration for which these spectra should occur needs to be defined.

To make use of this option, the user must specify a regular *instat* value for spectral wave boundary conditions (*jons*, *swan* or *vardens*), but instead of referencing a single spectrum definition file using the *bcfile* keyword, an extra file listing all spectrum definition files is now referenced.

The first word in this extra file must be the keyword *FILELIST*. In the following lines, each line contains the duration of this wave spectrum condition in seconds (similar to *rt* in *params.txt*), the required time step in this boundary condition file in seconds (similar to *dtbf* in *params.txt*) and the name of the spectral definition file used to generate these boundary conditions. The duration and boundary condition time step in this file overrules *rt* and *dtbf* in *params.txt*. XBeach does not reuse time-varying spectrum files. Therefore the total duration of all spectra should at least match the duration of the simulation.

A typical input file contains the following:

filelist.txt

FILELIST

1800 0.2 jonswap1.inp

1800 0.2 jonswap1.inp

1350 0.2 jonswap2.inp

1500 0.2 jonswap3.inp

1200 0.2 jonswap2.inp

3600 0.2 jonswap4.inp

Similar to time-varying spectral wave boundary conditions, also spatially varying wave boundary conditions can be defined using a similar construction. In order to apply spatially varying spectra on the offshore boundary, the user must specify set the keywords *wbcversion =3* and *nspectrumloc=ns* in *params.txt* where *ns* is the number of locations in which a spectrum is defined. By default the number of defined spectra is one.

Similar to time-varying spectral wave boundary conditions, its spatially varying sibling uses an extra file listing all relevant spectrum definition files. The first word in this extra file must be the keyword *LOCLIST*. This line should be followed by one line per spectrum definition location containing the world x-coordinate and world y-coordinate of the location that the input spectrum should apply, and the name of the file containing spectral wave information.

A typical input file for a run with three JONSWAP spectra contains the following:

loclist.txt

LOCLIST

0. 0. jonswap1.inp

0. 100. jonswap2.inp

0. 200. jonswap3.inp

Note that it is not possible to use a mix of JONSWAP, SWAN and variance density files in either a *FILELIST* or a *LOCLIST* construction. It is also not possible to vary *dthetaS\_XB* between files in case of non-nautical SWAN spectra. However, it is possible to combine *FILELIST* and *LOCLIST* files by referencing *FILELIST* files from the *LOCLIST* file. In this case all *FILELIST* files should adhere to the same time discretisation, so the duration and timestep values should be constant over al *FILELIST* files as well as the number of wave spectra definitions.

The manner in which a time series of short wave energy and bound long wave flux is calculated per offshore boundary point for spatially varying spectra is described in REF. The user is reminded that along the offshore boundary of the model, the wave energy, rather than the wave height, is interpolated linearly between input spectra without consideration of the physical aspects of the intermediate bathymetry. In cases with large gradients in wave energy, direction or period, the user should specify sufficient wave spectra for the model to accurately represent changes in offshore wave conditions.

### Notes on the generation of wave boundary conditions

At the start of the XBeach simulation, XBeach checks whether non-stationary varying wave boundary conditions are to be used. If this is the case, it next checks whether the wave spectrum of the wave boundary conditions is to change over time, or remain constant. If the wave spectrum is to remain constant, XBeach will only read from one input file to generate wave boundary conditions. If the wave spectrum is to vary in time, XBeach reads from multiple files.

Whether or not the wave spectrum of the boundary conditions changes over time, the XBeach module requires a record length during which the current wave spectral parameters are applied. For the duration of the record length, boundary conditions are calculated at every boundary condition file time step. These time steps are not required to be the same as the time steps in the XBeach main program; XBeach will interpolate where necessary. The boundary condition time steps should therefore only be small enough to accurately describe the incoming bound long waves. The statistical data for the generation of the wave boundary conditions is read from user-specified files. The XBeach module tapers the beginning and end of the boundary condition file. This is done to ensure smooth transitions from one boundary condition file to the next.

The combination of a large record length and a small time step lead to large demands on the system memory. If the memory requirement is too large, the user must choose to either enlarge the boundary condition time step, or to reduce the record length. In case of the latter, several boundary condition files can be generated and read sequentially. It is unwise however to reduce the record length too much, as then the transitions between the boundary condition files may affect the model results.

Every time the XBeach wave boundary condition module is run, it outputs data to the local directory. Metadata about the wave boundary conditions are stored in list files: *ebcflist.bcf* and *qbcflist.bcf*. The main XBeach program uses the list files to know how and when to read and generate boundary condition files. The actual incoming short-wave energy and long-wave mass flux data is stored in other files. These files have *E\_* and *q\_* prefixes. The main XBeach program uses these files for the actual forcing along the offshore edge.

## Tide and surge input

Files: timeseries

### Tide boundary conditions

| keyword | description | default | range | units | remark |
| --- | --- | --- | --- | --- | --- |
| paulrevere | Specifies tide on sea and land or two sea points if tideloc = 2 | land | land, sea |  |  |
| tideloc | Number of corner points on which a tide time series is specified | 0 | 0 - 4 | - |  |
| zs0file | Name of tide boundary condition series |  |  | <file> |  |

### Flow boundary conditions

Files: bc/gen.ezs (ref from stationary wbc)

| keyword | description | default | range | units | remark |
| --- | --- | --- | --- | --- | --- |
| ARC\* | Switch for active reflection compensation at seaward boundary | 1 | 0 - 1 | - |  |
| back | Switch for boundary at bay side | abs\_2d | wall, abs\_1d, abs\_2d, wlevel |  |  |
| epsi\* | Ratio of mean current to time varying current through offshore boundary | -1.0 | -1.0 - 0.2 | - |  |
| freewave\* | Switch for free wave propagation 0 = use cg (default); 1 = use sqrt(gh) in instat = ts\_2 | 0 | 0 - 1 | - |  |
| front | Switch for seaward flow boundary | abs\_2d | abs\_1d, abs\_2d, wall, wlevel, nonh\_1d, waveflume |  |  |
| left | Switch for lateral boundary at ny+1 | neumann | neumann, wall, no\_advec, neumann\_v |  |  |
| nc\* | Smoothing distance for estimating umean (defined as nr of cells) | par%ny+1 | 1 - par%ny+1 | - |  |
| order\* | Switch for order of wave steering, 1 = first order wave steering (short wave energy only), 2 = second oder wave steering (bound long wave corresponding to short wave forcing is added) | 2.0 | 1.0 - 2.0 | - |  |
| right | Switch for lateral boundary at 0 | neumann | neumann, wall, no\_advec, neumann\_v |  |  |
| tidetype\* | Switch for offfshore boundary, velocity boundary or instant water level boundary | velocity | instant, velocity |  |  |

## Water level (dam break)

Files: timeseries

| keyword | description | default | range | units | remark |
| --- | --- | --- | --- | --- | --- |
| hotstartflow\* | Switch for hotstart flow conditions with pressure gradient balanced by wind and bed stress | 0 | 0 - 1 | - |  |
| zs0 | Inital water level | 0.0 | -5.0 - 5.0 | m |  |
| zsinitfile | Name of inital water level file |  |  | <file> |  |

## Wind input

Files: timeseries

| keyword | description | default | range | units | remark |
| --- | --- | --- | --- | --- | --- |
| Cd\* | Wind drag coefficient | 0.002 | 0.0001 - 0.01 | - |  |
| rhoa\* | Air density | 1.25 | 1.0 - 2.0 | kgm^-3 |  |
| windfile | Name of file with non-stationary wind data |  |  | <file> |  |
| windth | Nautical wind direction, in case of stationary wind | 270.0 | -360.0 - 360.0 | deg |  |
| windv | Wind velocity, in case of stationary wind | 0.0 | 0.0 - 200.0 | ms^-1 |  |

## Sediment input

Files: fractions, layers

| keyword | description | default | range | units | remark |
| --- | --- | --- | --- | --- | --- |
| D15 | D15 grain size per grain type | 1 | size(par%D15 - None | m |  |
| D50 | D50 grain size per grain type | 1 | size(par%D50 - None | m |  |
| D90 | D90 grain size per grain type | 1 | size(par%D90 - None | m |  |
| dzg1\* | Thickness of top sediment class layers | par%dzg1 | 0.01 - 1.0 | m |  |
| dzg2\* | Nominal thickness of variable sediment class layer | par%dzg1 | 0.01 - 1.0 | m |  |
| dzg3\* | Thickness of bottom sediment class layers | par%dzg1 | 0.01 - 1.0 | m |  |
| nd\* | Number of computational layers in the bed | 3 | 3 - 1000 | - |  |
| ngd | Number of sediment classes | 1 | 1 - 20 | - |  |
| por | Porosity | 0.4 | 0.3 - 0.5 | - |  |
| rhos | Solid sediment density (no pores) | 2650.0 | 2400.0 - 2800.0 | kgm^-3 |  |
| sedcal\* | Sediment transport calibration coefficient per grain type | 1 | size(par%sedcal - None | - |  |
| ucrcal\* | Critical velocity calibration coefficient per grain type | 1 | size(par%ucrcal - None | - |  |

## Vegetation input

Files: species, maps

| keyword | description | default | range | units | remark |
| --- | --- | --- | --- | --- | --- |
| nveg | Number of vegetation species | -123 |  | - |  |
| veggiefile | Name of vegetation species list file |  |  | <file> |  |
| veggiemapfile | Name of vegetation species map file |  |  | <file> |  |

## Discharge input

Files: timeseries

| keyword | description | default | range | units | remark |
| --- | --- | --- | --- | --- | --- |
| disch\_loc\_file\* | Name of discharge locations file |  |  | <file> |  |
| disch\_timeseries\_file\* | Name of discharge timeseries file |  |  | <file> |  |
| ndischarge\* | Number of discharge locations | par%ndischarge | 0 - 100 | - |  |
| ntdischarge\* | Length of discharge time series | par%ntdischarge | 0 - 100 | - |  |

## Drifters input

Files: timeseries

| keyword | description | default | range | units | remark |
| --- | --- | --- | --- | --- | --- |
| drifterfile | Name of drifter data file |  |  | <file> |  |
| ndrifter | Number of drifers | par%ndrifter | 0 - 50 | - |  |

## Ship input

Files: timeseries

| keyword | description | default | range | units | remark |
| --- | --- | --- | --- | --- | --- |
| nship\* | Number of ships | -123 |  | - |  |
| shipfile | Name of ship data file |  |  | <file> |  |

## Output selection

Files: output

| keyword | description | default | range | units | remark |
| --- | --- | --- | --- | --- | --- |
| globalvars\* | Mnems of global output variables, not per se the same size as nglobalvar (invalid variables, defaults) | 'abc' |  | - |  |
| meanvars\* | Mnems of mean output variables (by variables) | 'abc' |  | - |  |
| ncfilename\* | Xbeach netcdf output file name |  |  | <file> |  |
| ncross\* | Number of output cross sections | 0 | 0 - 50 | - |  |
| nglobalvar | Number of global output variables (as specified by user) | -1 | -1 - 20 | - |  |
| nmeanvar | Number of mean, min, max, var output variables | 0 | 0 - 15 | - |  |
| npoints | Number of output point locations | 0 | 0 - 50 | - |  |
| npointvar | Number of point output variables | 0 | 0 - 50 | - |  |
| nrugauge | Number of output runup gauge locations | 0 | 0 - 50 | - |  |
| nrugdepth\* | Number of depths to compute runup in runup gauge | 1 | 1 - 10 | - |  |
| outputformat\* | Output file format | fortran | fortran, netcdf, debug |  |  |
| pointtypes\* | Point types (0 = point, 1 = rugauge) | > NULL() |  | - |  |
| pointvars\* | Mnems of point output variables (by variables) | 'abc' |  | - |  |
| rugdepth\* | Minimum depth for determination of last wet point in runup gauge | 1 | size(par%rugdepth - None | m |  |
| timings\* | Switch enable progress output to screen | 1 | 0 - 1 | - |  |
| tintc\* | Interval time of cross section output | 1.0 | 0.01 - 100000.0 | s |  |
| tintg | Interval time of global output | 1.0 | 0.01 - 100000.0 | s |  |
| tintm | Interval time of mean, var, max, min output | par%tstop-par%tstart | 1.0 - par%tstop-par%tstart | s |  |
| tintp | Interval time of point and runup gauge output | 1.0 | 0.01 - 100000.0 | s |  |
| tscross\* | Name of file containing timings of cross section output | None | None - None | - |  |
| tsglobal\* | Name of file containing timings of global output | None | None - None | - |  |
| tsmean\* | Name of file containing timings of mean, max, min and var output | None | None - None | - |  |
| tspoints\* | Name of file containing timings of point output | None | None - None | - |  |
| tstart | Start time of output, in morphological time | 1.0 | 0.0 - 1000000.0 | s |  |

## Time parameters

Files: timings

| keyword | description | default | range | units | remark |
| --- | --- | --- | --- | --- | --- |
| CFL | Maximum Courant-Friedrichs-Lewy number | 0.7 | 0.1 - 0.9 | - |  |
| tstop | Stop time of simulation, in morphological time | 2000.0 | 1.0 - 1000000.0 | s |  |
| tunits\* | Time units in udunits format (seconds since 1970-01-01 00:00:00.00 +1:00) | 's' |  | - |  |

## Model coefficients

### Wave numerics

| keyword | description | default | range | units | remark |
| --- | --- | --- | --- | --- | --- |
| maxerror\* | Maximum wave height error in wave stationary iteration | 5e-05 | 1e-05 - 0.001 | m |  |
| maxiter\* | Maximum number of iterations in wave stationary | 500 | 2 - 1000 | - |  |
| scheme\* | Numerical scheme for wave propagation | upwind\_2 | upwind\_1, lax\_wendroff, upwind\_2 |  |  |
| wavint | Interval between wave module calls (only in stationary wave mode) | 60.0 | 1.0 - 3600.0 | s |  |

### Wave dissipation

| keyword | description | default | range | units | remark |
| --- | --- | --- | --- | --- | --- |
| alpha\* | Wave dissipation coefficient in Roelvink formulation | 1.0 | 0.5 - 2.0 | - |  |
| break | Type of breaker formulation | roelvink2 | roelvink1, baldock, roelvink2, roelvink\_daly, janssen |  |  |
| breakerdelay\* | Switch to enable breaker delay model | 1 | 0 - 1 | - |  |
| delta\* | Fraction of wave height to add to water depth | 0.0 | 0.0 - 1.0 | - |  |
| facrun\* | Calibration coefficient for short wave runup | 1.0 | 0.0 - 2.0 | - |  |
| facsd\* | Fraction of the local wave length to use for shoaling delay depth | 1.0 | 0.0 - 2.0 | - |  |
| fw\* | Bed friction factor | 0.0 | 0.0 - 1.0 | - |  |
| fwcutoff | Depth greater than which the bed friction factor is not applied | 1000.0 | 0.0 - 1000.0 | - |  |
| gamma | Breaker parameter in Baldock or Roelvink formulation | 0.55 | 0.4 - 0.9 | - |  |
| gamma2 | End of breaking parameter in Roelvink Daly formulation | 0.3 | 0.0 - 0.5 | - |  |
| gammax\* | Maximum ratio wave height to water depth | 2.0 | 0.4 - 5.0 | - |  |
| n\* | Power in Roelvink dissipation model | 10.0 | 5.0 - 20.0 | - |  |
| shoaldelay\* | Switch to enable shoaling delay | 0 | 0 - 1 | - |  |

### Rollers

| keyword | description | default | range | units | remark |
| --- | --- | --- | --- | --- | --- |
| beta\* | Breaker slope coefficient in roller model | 0.1 | 0.05 - 0.3 | - |  |
| rfb\* | Switch to feed back maximum wave surface slope in roller energy balance, otherwise rfb = par%Beta | 0 | 0 - 1 | - |  |
| roller\* | Switch to enable roller model | 1 | 0 - 1 | - |  |

### Wave-current interaction

| keyword | description | default | range | units | remark |
| --- | --- | --- | --- | --- | --- |
| cats\* | Current averaging time scale for wci, in terms of mean wave periods | 4.0 | 1.0 - 50.0 | Trep |  |
| hwci\* | Minimum depth until which wave-current interaction is used | 0.1 | 0.001 - 1.0 | m |  |
| wci | Turns on wave-current interaction | 0 | 0 - 1 | - |  |

### Flow

| keyword | description | default | range | units | remark |
| --- | --- | --- | --- | --- | --- |
| C | Chezy coefficient | 55.0 | 20.0 - 100.0 | m^0.5s^-1 |  |
| bedfriccoef | Bed friction coefficient | 0.01 | 3.5e-05 - 0.9 | - |  |
| bedfricfile | Bed friction file (only valid with values of C) |  |  | <file> |  |
| bedfriction | Bed friction formulation | chezy | chezy, cf, white-colebrook, manning, white-colebrook-grainsize |  |  |
| cf\* | Friction coefficient flow | 0.003 | 0.001 - 0.1 | - |  |
| nuh | Horizontal background viscosity | 0.1 | 0.0 - 1.0 | m^2s^-1 |  |
| nuhfac\* | Viscosity switch for roller induced turbulent horizontal viscosity | 1.0 | 0.0 - 1.0 | - |  |
| nuhv\* | Longshore viscosity enhancement factor, following Svendsen (?) | 1.0 | 1.0 - 20.0 | - |  |
| smag\* | Switch for smagorinsky subgrid model for viscocity | 1 | 0 - 1 | - |  |

### Flow numerics

| keyword | description | default | range | units | remark |
| --- | --- | --- | --- | --- | --- |
| eps | Threshold water depth above which cells are considered wet | 0.005 | 0.001 - 0.1 | m |  |
| eps\_sd | Threshold velocity difference to determine conservation of energy head versus momentum | 0.5 | 0.0 - 1.0 | m/s |  |
| hmin | Threshold water depth above which Stokes drift is included | 0.2 | 0.001 - 1.0 | m |  |
| oldhu\* | Switch to enable old hu calculation | 0 | 0 - 1 | - |  |
| secorder\* | Use second order corrections to advection/non-linear terms based on MacCormack scheme | 0 | 0 - 1 | - |  |
| umin | Threshold velocity for upwind velocity detection and for vmag2 in equilibrium sediment concentration | 0.0 | 0.0 - 0.2 | m/s |  |

### Sediment transport

| keyword | description | default | range | units | remark |
| --- | --- | --- | --- | --- | --- |
| BRfac\* | Calibration factor surface slope | 1.0 | 0.0 - 1.0 | - |  |
| Tbfac\* | Calibration factor for bore interval Tbore: Tbore = Tbfac\*Tbore | 1.0 | 0.0 - 1.0 | - |  |
| Tsmin\* | Minimum adaptation time scale in advection diffusion equation sediment | 0.5 | 0.01 - 10.0 | s |  |
| bdslpeffdir | Modify the direction of the sediment transport based on the bed slope | none | none, talmon |  |  |
| bdslpeffdirfac | Calibration factor in the modification of the direction | 1.0 | 0.0 - 2.0 | - |  |
| bdslpeffini | Modify the critical shields parameter based on the bed slope | none | none, total, bed |  |  |
| bdslpeffmag | Modify the magnitude of the sediment transport based on the bed slope, uses facsl | roelvink\_total | none, roelvink\_total, roelvink\_bed, soulsby\_total, soulsby\_bed |  |  |
| bed\* | Calibration factor for bed transports | 1 | 0 - 1 | - |  |
| betad\* | Dissipation parameter long wave breaking turbulence | 1.0 | 0.0 - 10.0 | - |  |
| bulk\* | Switch to compute bulk transport rather than bed and suspended load separately | 0 | 0 - 1 | - |  |
| dilatancy | Switch to reduce critical shields number due dilatancy | 0 | 0 - 1 | - |  |
| facAs\* | Calibration factor time averaged flows due to wave asymmetry | 0.1 | 0.0 - 1.0 | - |  |
| facDc\* | Option to control sediment diffusion coefficient | 1.0 | 0.0 - 1.0 | - |  |
| facSk\* | Calibration factor time averaged flows due to wave skewness | 0.1 | 0.0 - 1.0 | - |  |
| facsl\* | Factor bedslope effect | 1.6 | 0.0 - 1.6 | - |  |
| facua\* | Calibration factor time averaged flows due to wave skewness and asymmetry | 0.1 | 0.0 - 1.0 | - |  |
| fallvelred | Switch to reduce fall velocity for high concentrations | 0 | 0 - 1 | - |  |
| form | Equilibrium sediment concentration formulation | vanthiel\_vanrijn | soulsby\_vanrijn, vanthiel\_vanrijn |  |  |
| jetfac\* | Option to mimic turbulence production near revetments | 0.0 | 0.0 - 1.0 | - |  |
| lws\* | Switch to enable long wave stirring | 1 | 0 - 1 | - |  |
| lwt\* | Switch to enable long wave turbulence | 0 | 0 - 1 | - |  |
| pormax | Max porosity used in the experession of Van Rhee | 0.5 | 0.3 - 0.6 | - |  |
| reposeangle | Angle of internal friction | 30.0 | 0.0 - 45.0 | deg |  |
| rheeA | A parameter in the Van Rhee expression | 0.75 | 0.75 - 2.0 | - |  |
| smax\* | Maximum Shields parameter for equillibrium sediment concentration acc. Diane Foster | -1.0 | -1.0 - 3.0 | - |  |
| sus\* | Calibration factor for suspensions transports | 1 | 0 - 1 | - |  |
| sws\* | Switch to enable short wave and roller stirring and undertow | 1 | 0 - 1 | - |  |
| tsfac\* | Coefficient determining Ts = tsfac \* h/ws in sediment source term | 0.1 | 0.01 - 1.0 | - |  |
| turb\* | Switch to include short wave turbulence | bore\_averaged | none, wave\_averaged, bore\_averaged |  |  |
| turbadv\* | Switch to activate turbulence advection model for short and or long wave turbulence | none | none, lagrangian, eulerian |  |  |
| waveform | Wave shape model | vanthiel | ruessink\_vanrijn, vanthiel |  |  |
| z0\* | Zero flow velocity level in Soulsby and van Rijn (1997) sediment concentration | 0.006 | 0.0001 - 0.05 | m |  |

### Sediment transport numerics

| keyword | description | default | range | units | remark |
| --- | --- | --- | --- | --- | --- |
| cmax\* | Maximum allowed sediment concentration | 0.1 | 0.0 - 1.0 | - |  |
| sourcesink\* | Switch to enable source-sink terms to calculate bed level change rather than suspended transport gradients | 0 | 0 - 1 | - |  |
| thetanum\* | Coefficient determining whether upwind (1) or central scheme (0.5) is used. | 1.0 | 0.5 - 1.0 | - |  |

### Quasi-3D sediment transport

| keyword | description | default | range | units | remark |
| --- | --- | --- | --- | --- | --- |
| kmax\* | Number of sigma layers in Quasi-3D model; kmax = 1 is without vertical structure of flow and suspensions | 1 | 1 - 1000 | - |  |
| sigfac\* | Dsig scales with log(sigfac) | 1.3 | 0.0 - 10.0 | - |  |
| vicmol\* | Molecular viscosity | 1e-06 | 0.0 - 0.001 | - |  |
| vonkar\* | Von Karman constant | 0.4 | 0.01 - 1.0 | - |  |

### Morphology

| keyword | description | default | range | units | remark |
| --- | --- | --- | --- | --- | --- |
| dryslp | Critical avalanching slope above water (dz/dx and dz/dy) | 1.0 | 0.1 - 2.0 | - |  |
| dzmax\* | Maximum bed level change due to avalanching | 0.05 | 0.0 - 1.0 | m/s/m |  |
| hswitch\* | Water depth at which is switched from wetslp to dryslp | 0.1 | 0.01 - 1.0 | m |  |
| morfac | Morphological acceleration factor | 1.0 | 0.0 - 1000.0 | - |  |
| morfacopt\* | Switch to adjusting output times for morfac | 1 | 0 - 1 | - |  |
| morstart | Start time morphology, in morphological time | 120.0 | 0.0 - 10000000.0 | s |  |
| morstop | Stop time morphology, in morphological time | 2000.0 | 0.0 - 10000000.0 | s |  |
| ne\_layer | Name of file containing depth of hard structure |  |  | <file> |  |
| struct | Switch for enabling hard structures | 0 | 0 - 1 | - |  |
| wetslp | Critical avalanching slope under water (dz/dx and dz/dy) | 0.3 | 0.1 - 1.0 | - |  |

### Bed update

| keyword | description | default | range | units | remark |
| --- | --- | --- | --- | --- | --- |
| frac\_dz\* | Relative thickness to split time step for bed updating | 0.7 | 0.5 - 0.98 | - |  |
| merge\* | Merge threshold for variable sediment layer (ratio to nominal thickness) | 0.01 | 0.005 - 0.1 | - |  |
| nd\_var\* | Index of layer with variable thickness | 2 | 2 - par%nd | - |  |
| nsetbathy\* | Number of prescribed bed updates | 1 | 1 - 1000 | - |  |
| setbathyfile\* | Name of prescribed bed update file |  |  | <file> |  |
| split\* | Split threshold for variable sediment layer (ratio to nominal thickness) | 1.01 | 1.005 - 1.1 | - |  |

### Groundwater flow

| keyword | description | default | range | units | remark |
| --- | --- | --- | --- | --- | --- |
| aquiferbot\* | Level of uniform aquifer bottom | -10.0 | -100.0 - 100.0 | m |  |
| aquiferbotfile\* | Name of the aquifer bottom file |  |  | <file> |  |
| dwetlayer\* | Thickness of the top soil layer interacting more freely with the surface water | 0.1 | 0.01 - 1.0 | m |  |
| gw0\* | Level initial groundwater level | 0.0 | -5.0 - 5.0 | m |  |
| gw0file\* | Name of initial groundwater level file |  |  | <file> |  |
| gwReturb\* | Reynolds number for start of turbulent flow in case of gwscheme = turbulent | 100.0 | 1.0 - 600.0 | - |  |
| gwfastsolve\* | Reduce full 2D non-hydrostatic solution to quasi-explicit in longshore direction | 0 | 0 - 1 | - |  |
| gwheadmodel\* | Model to use for vertical groundwater head | parabolic | parabolic, exponential |  |  |
| gwhorinfil\* | Switch to include horizontal infiltration from surface water to groundwater | 0 | 0 - 1 | - |  |
| gwnonh\* | Switch to turn on or off non-hydrostatic pressure for groundwater | 0 | 0 - 1 | - |  |
| gwscheme\* | Scheme for momentum equation | laminar | laminar, turbulent |  |  |
| kx\* | Darcy-flow permeability coefficient in x-direction | 0.0001 | 1e-05 - 0.1 | ms^-1 |  |
| ky\* | Darcy-flow permeability coefficient in y-direction | 0.0001 | 1e-05 - 0.1 | ms^-1 |  |
| kz\* | Darcy-flow permeability coefficient in z-direction | 0.0001 | 1e-05 - 0.1 | ms^-1 |  |

### Non-hydrostatic correction

| keyword | description | default | range | units | remark |
| --- | --- | --- | --- | --- | --- |
| Topt\* | Absolute period to optimize coefficient | 10.0 | 1.0 - 20.0 | s |  |
| breakviscfac\* | Factor to increase viscosity during breaking | 1.5 | 1.0 - 3.0 | - |  |
| breakvisclen\* | Ratio between local depth and length scale in extra breaking viscosity | 1.0 | 0.75 - 3.0 | - |  |
| dispc\* | Coefficient in front of the vertical pressure gradient | 1.0 | 0.1 - 2.0 | ? |  |
| kdmin\* | Minimum value of kd (pi/dx > min(kd)) | 0.0 | 0.0 - 0.05 | - |  |
| maxbrsteep\* | Maximum wave steepness criterium | 0.6 | 0.3 - 0.8 | - |  |
| nhbreaker\* | Non-hydrostatic breaker model | 2 | 0 - 3 | - |  |
| reformsteep\* | Wave steepness criterium to reform after breaking | 0.25d0\*par%maxbrsteep | 0.0 - 0.95d0\*par%maxbrsteep | - |  |
| secbrsteep\* | Secondary maximum wave steepness criterium | 0.5d0\*par%maxbrsteep | 0.0 - 0.95d0\*par%maxbrsteep | - |  |
| solver\* | Solver used to solve the linear system | tridiag | sip, tridiag |  |  |
| solver\_acc\* | Accuracy with respect to the right-hand side usedin the following termination criterion: ||b-Ax || < acc\*||b|| | 0.005 | 1e-05 - 0.1 | - |  |
| solver\_maxit\* | Maximum number of iterations in the linear sip solver | 30 | 1 - 1000 | - |  |
| solver\_urelax\* | Underrelaxation parameter | 0.92 | 0.5 - 0.99 | - |  |

### Physical constants

| keyword | description | default | range | units | remark |
| --- | --- | --- | --- | --- | --- |
| depthscale\* | Depthscale of (lab)test simulated, affects eps, hmin, hswitch and dzmax | 1.0 | 1.0 - 200.0 | - |  |
| g | Gravitational acceleration | 9.81 | 9.7 - 9.9 | ms^-2 |  |
| rho | Density of water | 1025.0 | 1000.0 - 1040.0 | kgm^-3 |  |

### Coriolis force

| keyword | description | default | range | units | remark |
| --- | --- | --- | --- | --- | --- |
| lat\* | Latitude at model location for computing Coriolis | 0.0 | -90.0 - 90.0 | deg |  |
| wearth\* | Angular velocity of earth calculated as: 1/rotation\_time (in hours) | 1.d0/24.d0 | 0.0 - 1.0 | hour^-1 |  |

### MPI

| keyword | description | default | range | units | remark |
| --- | --- | --- | --- | --- | --- |
| mmpi\* | Number of domains in cross-shore direction when manually specifying mpi domains | 2 | 1 - 100 | - |  |
| mpiboundary\* | Fix mpi boundaries along y-lines, x-lines, use manual defined domains or find shortest boundary automatically | auto | auto, x, y, man |  |  |
| nmpi\* | Number of domains in alongshore direction when manually specifying mpi domains | 4 | 1 - 100 | - |  |

### Output projection

| keyword | description | default | range | units | remark |
| --- | --- | --- | --- | --- | --- |
| projection\* | Projection string | ' ' |  | - |  |
| rotate | Rotate output as postprocessing with given angle | 1 | 0 - 1 | - |  |
|  |  |  |  |  |  |

# Bibliography

Holthuijsen, L., Booij, N., & Herbers, T. (1989). A prediction model for stationary, short-crested waves in shallow water with ambient currents. Coastal Engineering, 13(1):23-54.

Roelvink, J.A. (1993a) Dissipation in random wave groups incident on a beach. Coastal Engineering, pp. 127-150.

Roelvink, J.A. (1993b) Surf beat and its effect on cross-shore profiles. Ph.D. Thesis, Delft University of Technology.

Daly, C., Roelvink, J. A., Van Dongeren, A., Van Thiel de Vries, J. S. M., & McCall, R. (2010). Short wave breaking effects on low frequency waves. Proceedings of 32nd International Conference on Coastal Engineering, (1), 1–13.

# Tutorial

Nog niet verdeeld. Later nog in te vullen.

## 1-D profile model

Delfland Deltagoot

## 2-D area model

Ocean bay park: getij+surge, baai, duin, nonerodible, overwash, collision,

## Langsgetij + riveroutflow

getijmodel + rivier + stationair.