windsurf Documentation

Release 0.1

Bas Hoonhout

July 27, 2016

Contents

CHAPTER 1

Contents

1.1 The Windsurf model

1.1.1 What is it?

The Windsurf model is a composite model that connects three different model cores for simulating integrated nearshore and aeolian sediment transport. The Windsurf model connects the following model cores:

- XBeach - Nearshore hydrodynamics (http://www.xbeach.org)
- AeoLiS - Supply-limited aeolian sediment transport (http://openearth.github.io/aeolis/)
- Coastal Dune Model (CDM) - Aeolian sediment transport and wind shear accounting for morphological feedback and vegetation

The Windsurf model simulates 2DH nearshore and aeolian sediment transport as a result of short waves, infragravity waves, tides and currents and wind. The Windsurf model accounts for multiple sediment fractions and bed layers, sediment supply limitations in aeolian transport as a result of moisture contents, sediment sorting and beach armoring, morphological feedback and vegetation.

1.1.2 How to use it?

The Windsurf composite model features a Python interface that connects the three different model cores and acts as a user-interface for the end-user. The Python interface can be downloaded as Python package from the OpenEarth GitHub respository: https://github.com/openearth/windsurf/.

The installation and configuration of a Windsurf model is described in the following subsections.

Installation

Download the individual model cores from their respective repositories and compile the models as libraries according to their manuals:

- XBeach: https://svn.oss.deltares.nl/repos/xbeach/branches/fedor-template/ (at this moment the XBeach trunk does not include the necessary BMI interface, use the "fedor-template" branch instead)
- AeoLiS: https://github.com/openearth/windsurf/
- CDM: ?

Download the Windsurf Python package from https://github.com/openearth/windsurf/ and install using:

```
>>> python setup.py install
```

Check if the installation is successful using:

```
>>> windsurf --help
windsurf : a composite model for simulating integrated nearshore and aeolian sediment transport

Usage:
    windsurf <config> [--verbose=LEVEL]

Positional arguments:
    config              configuration file

Options:
    -h, --help          show this help message and exit
    --verbose=LEVEL     print logging messages [default: 30]
```

Configuration

The Windsurf model is configured through a single JSON file. The JSON file contains different categories of configuration options that are treated in this section. A JSON configuration file may contain the following:

```
{
    "time" : {
        "start" : 0.0,
        "stop"  : 31536000.0
    },
    "models" : {
        "xbeach" : {
            "engine" : "xbeachmi.model.XBeachMI",
            "engine_path" : "/Users/hoonhout/Checkouts/XBeach/trunk/src/xbeachlibrary/.libs/",
            "configfile" : "xbeachmi.json"
        },
        "aeolis" : {
            "engine" : "aeolis",
            "engine_path" : "/Users/hoonhout/Github/aeolis/src/.libs/",
            "configfile" : "aeolis.txt"
        },
        "cdm" : {
            "engine" : "cdm",
            "engine_path" : "/Users/hoonhout/Github/cdm/.libs/",
            "configfile" : "cdm.txt"
        }
    },
    "exchange" : [
        {
            "var_from" : "xbeach.zb",
            "var_to" : "aeolis.zbx"
        },{
            "var_from" : "xbeach.zs",
            "var_to" : "aeolis.zs"
        },{
            "var_from" : "xbeach.H",
            "var_to" : "aeolis.Hs"
        },{
            "var_from" : "aeolis.zb",
            "var_to" : "xbeach.zb"
```

```
        },{
            "var_from" : "aeolis.zs",
            "var_to" : "xbeach.zs"
        }
    ],
    "regimes" : {
        "stat" : {
            "xbeach" : {
                "instance" : "stat"
            },
            "aeolis" : {
                "scheme" : "euler_backward",
                "dt" : 60.0,
                "accfac" : 10.0
            }
        },
        "instat" : {
            "xbeach" : {
                "instance" : "instat"
            },
            "aeolis" : {
                "scheme" : "euler_backward",
                "dt" : 60.0,
                "accfac" : 10.0
            }
        }
    },
    "scenario" : [
        [0.0, "stat"],
        [730800.0, "instat"],
        [1004400.0, "stat"],
        [1890000.0, "instat"],
        [2012400.0, "stat"],
        [2239200.0, "instat"],
        [2466000.0, "stat"],
        [2815200.0, "instat"],
        [3265200.0, "stat"],
        [3826800.0, "instat"],
        [3938400.0, "stat"],
        [4352400.0, "instat"],
        [4410000.0, "stat"],
        [5115600.0, "instat"],
        [5256000.0, "stat"],
        [5612400.0, "instat"],
        [5821200.0, "stat"],
        [6451200.0, "instat"],
        [6537600.0, "stat"],
        [6973200.0, "instat"],
        [7358400.0, "stat"],
        [7491600.0, "instat"],
        [7632000.0, "stat"],
        [7783200.0, "instat"],
        [7887600.0, "stat"],
        [8161200.0, "instat"],
        [8276400.0, "stat"],
        [8640000.0, "instat"],
        [8744400.0, "stat"],
        [10047600.0, "instat"],
```

```
        [10155600.0, "stat"],
        [10674000.0, "instat"],
        [10767600.0, "stat"],
        [10990800.0, "instat"],
        [11959200.0, "stat"],
        [19112400.0, "instat"],
        [19292400.0, "stat"],
        [24256800.0, "instat"],
        [24433200.0, "stat"],
        [27082800.0, "instat"],
        [27183600.0, "stat"],
        [31536000.0, "instat"]
    ],
    "restart" : {
        "variables" : ["xbeach.zb","xbeach.Fx","xbeach.Fy","xbeach.Sxy","xbeach.Syy","xbeach.Sxx","xb
        "times" : [86400.0, 172800.0, 259200.0, 2678400.0, 5270400.0, 7948800.0, 10627200.0, 1304640
        "backup" : true
    },
    "netcdf" : {
        "outputfile" : "windsurf.nc",
        "outputvars" : ["zb", "zs", "H", "Ct.avg", "Cu.avg", "uw.avg", "uth.avg", "mass.avg", "supply
        "interval" : 3600.0,
        "crs" : {
            "grid_mapping_name" : "oblique_stereographic",
            "epsg_code" : "EPSG:28992",
            "semi_major_axis" : 6377397.155,
            "semi_minor_axis" : 6356078.96282,
            "inverse_flattening" : 299.1528128,
            "latitude_of_projection_origin" : 52.0922178,
            "longitude_of_projection_origin" : 5.23155,
            "scale_factor_at_projection_origin" : 0.9999079,
            "false_easting" : 155000.0,
            "false_northing" : 463000.0,
            "proj4_params" : "+proj=sterea +lat_0=52.15616055555555 +lon_0=5.38763888888889 +k=0.999
        },
        "attributes" : {
            "institution" : "Delft University of Technology",
            "creator_name" : "Bas Hoonhout",
            "creator_email" : "b.m.hoonhout@tudelft.nl"
        }
    }
}
```

time

Time management.

models

Model engine specification and configuration.

exchange

Data exchange between model engines.

regimes

Environmental regime specification and configuration.

scenario

Scenario configuration (sequence of regimes)

Execution

Execute the model by calling the following command from the command-line:

```
>>> windsurf windsurf.json
```

To print more output to the screen decrease the verbosity number as follows:

```
>>> windsurf windsurf.json --verbose=20
```

Tow write the output to a file use the following:

```
>>> windsurf windsurf.json --verbose=20 > windsurf.log
```

1.2 Source code documentation

The windsurf Python package connects the model cores within the Windsurf composite model and acts as user-interface to the end-user. The package consists of different modules that are documented in the following sections.

1.2.1 model

1.2.2 netcdf

1.2.3 parsers

class parsers.**AeolisParser**(*configfile*)
Configuration parser class for AeoLiS models

Inherits from ConfigParser.

class parsers.**ConfigParser**(*configfile*)
Configuration parser base class

Base class for the construction of model engine configuration file parsers. Parses the main configuration file and referenced files therin.

__init__(*configfile*)
Initialize the class

Parameters **configfile** (*str*) – path to model configuration file

parse()
Parse configuration file

Returns key/value pairs of model configuration

Return type dict

parse_config_file(*configfile*)
Parse configuration file

Parameters **configfile** (*str*) – path to configuration file

Returns key/value pairs of model configuration

Return type dict

parse_config_value(*value*, *force_list=False*)
Parse configuration value string to valid Python variable type

Parameters **value** (*str*) – configuration value string

Returns parsed configuration value

Return type str, int, float, bool or list

parse_referenced_file(*fname*)
Parse a file referenced in the main configuration file

Parameters **fname** (*str*) – referenced filename

Returns a np.ndarray for numeric data, a dictionary for key/value data or a list for plain text data

Return type np.ndarray, dict or list

class parsers.**XBeachParser**(*configfile*)
Configuration parser class for XBeach models

Inherits from ConfigParser.

1.2.4 console

1.3 Tutorials and examples

CHAPTER 2

Command-line tools

The Windsurf model can be executed from the command-line using the "windsurf" command. See for more information the *–help* option.

2.1 windsurf

CHAPTER 3

Source code repository

The Windsurf source code can be downloaded from the OpenEarth GitHub repository: https://github.com/openearth/windsurf/.

CHAPTER 4

Acknowledgements

The Windsurf model is an initiated by:

- Oregon State University
  - Peter Ruggiero
  - Nick Cohn
- University of North Carolina
  - Laura Moore
  - Evan Goldstein
- Delft University of Technology
  - Bas Hoonhout
  - Sierd de Vries
- Bremen University
  - Orencio Durán
- UNESCO-IHE
  - Dano Roelvink

The Python package is developed and currently maintained by Bas Hoonhout.

Bas Hoonhout and Sierd de Vries are supported by the ERC-Advanced Grant 291206 Nearshore Monitoring and Modeling (NEMO) and Deltares for their work on the Windsurf model.

Peter Ruggiero, Nick Cohn, Laura Moore and Evan Goldstein are supported by ... for their work on the Windsurf model.

CHAPTER 5

Indices and tables

- *genindex*
- *modindex*
- *search*

p

Symbols

A

C

P

X