Towards tractable sea level simulations

by

George Breyiannis

European Commission - Joint Research Centre

Context

- Unit: JRC.E.1 - Disaster Risk Management Unit
- Copernicus Emergency Management Service :https://emergency.copernicus.eu
- Floods Team:
  - **EFAS:https://www.efas.eu**
  - **GLOFAS: https://www.globalfloods.eu**
- Coastal flooding component : Pre-Operational system for computing the hazard, evaluating exposure assessing the impact, analyzing compound events, etc.

European Commission

Challenge

- To develop a global integrated coastal flood risk management system that links
- Satellite monitoring
- Coupled wave, tide and surge forecasting
- Inundation modelling
- Impact analysis.

Expected output

- (i) model coastal extreme water levels.
- (ii) derive joint return periods and inundation for concurring inland and coastal floods.
- (iii) calculate coastal flood impacts.

pyPoseidon

- A python framework for Hydrodynamic simulations

  (https://github.com/brey/pyPoseidon)

Desired attributes

- Pre/Post processing
- Multiple solvers
- Reproducibility
- Transparency
- Portability
- Scalability
- Interoperability

Features

- Handles Pre/Post processing (grid generation, automatic dataset retrieval and alignment, model setup, validation, etc.)
- Reproducibility (Extensive logging, run signature, etc.)
- Transparency (self contained in a package, open source, documented, etc.)
- Expandability (open ended in features)
- Portability (cross platform with conda & python)
- Scalability (scales from portable computer to HPC)
- Interoperability (based on ever an expanding set of popular python structures like pandas, xarray etc.)

Structure

- model
- **d3d** (http://oss.deltares.nl/web/delft3d/source-code)
- **schism** (http://ccrm.vims.edu/schismweb/)
- grid
- **jigsaw** (https://github.com/dengwirda/jigsaw)
- **ugmsh** (http://gmsh.info)
- dem
- meteo
- utils
- *tide*
- misc

DEMO

```
import pyPoseidon
import os
DATA_PATH = os.path.dirname(pyPoseidon.__file__)
pyPoseidon.__version__
```

```
## Minimal test
case = {
    'solver':'schism',
    'geometry':{
        'lon_min' : -30.,
        'lon_max' : -10.,
        'lat_min' : 60.,
        'lat_max' : 70.},
     'start_date':'2017-10-1 0:0:0',
     'time_frame':'12H',
     'meteo_source' : [DATA_PATH + '/tests/data/erai.grib']
     'meteo_engine' : 'cfgrib',
     'dem_source': DATA_PATH + '/tests/data/dem.nc'
       }
```

```
b = pyPoseidon.model(**case)
b.execute()
```

```
INFO:pyPoseidon:Creating grid with JIGSAW

INFO:pyPoseidon:Creating JIGSAW files

INFO:pyPoseidon:executing jigsaw

INFO:pyPoseidon:Jigsaw FINISHED

INFO:pyPoseidon:..reading mesh

INFO:pyPoseidon:..done creating mesh
```

```
b.get_data()
```

```
INFO:pyPoseidon:Combining output for folder ./schism/

INFO:pyPoseidon:Retrieve station timeseries if any

INFO:pyPoseidon:no station data loaded
INFO:pyPoseidon:Retrieve observations info
```

```
b.data.Dataset
```

xarray.Dataset

▶ **Dimensions:** (nMaxSCHISM_hgrid_face_nodes: 3, nSCHISM_hgrid_edge: 3182, nSCHISM_hgrid_face: 1956, nSCHISM_hgrid_node: 1226, nSCHISM_vgrid_layers: 2, one: 1, **sigma**: 2, **time**: 12, two: 2)

▼ **Coordinates:**

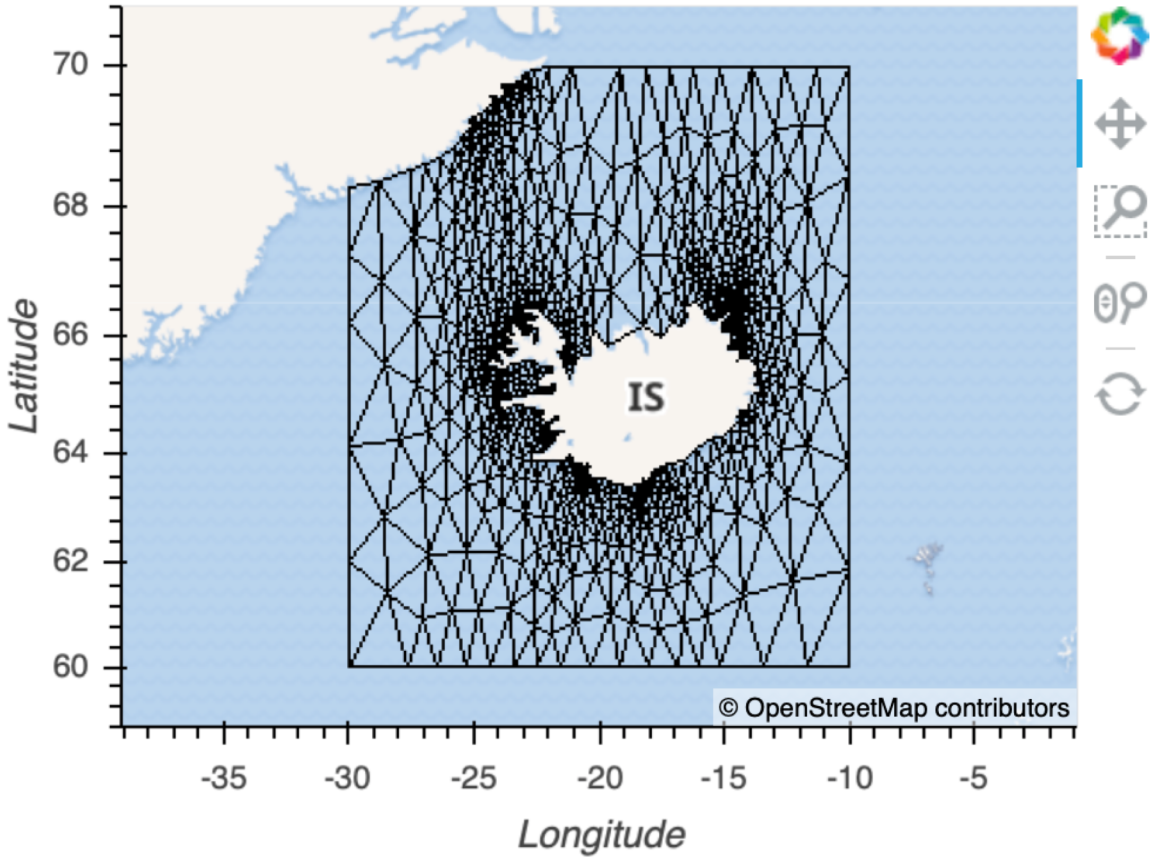
time	(time)	datetime64[ns]	2017-10-01T01:00:00 ... 2017-10-...
sigma	(sigma)	float64	-1.0 0.0

▶ **Data variables:** (25)

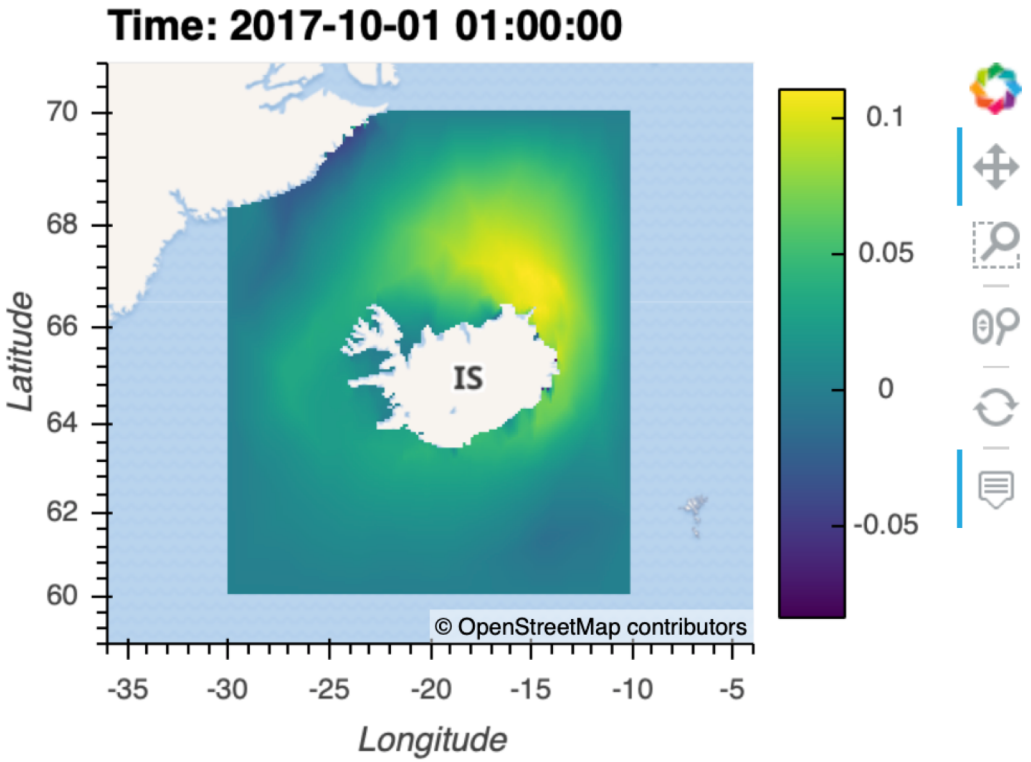
▼ **Attributes:**

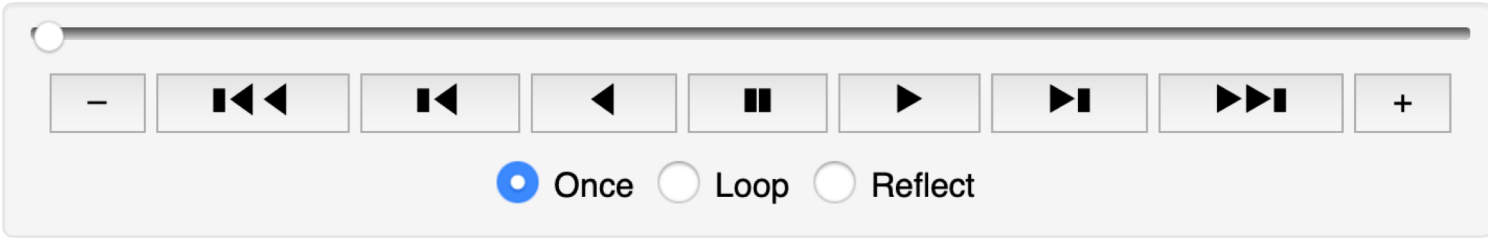
Conventions :	CF-1.0, UGRID-1.0
title :	SCHISM Model output
source :	SCHISM model output version v10
references :	http://ccrm.vims.edu/schismweb/
history :	created by pyPoseidon
comment :	SCHISM Model output
type :	SCHISM Model output
VisIT_plugin :	https://schism.water.ca.gov/library/-/document_library/view/3476283

```
b.data.Dataset.hplot.grid(width=400,height=300)
```



```
b.data.Dataset.hplot.frames(var='elev', width=400, height=300)
```





```
#### Simulation signature is in the json file

!ls schism/
```

```
bctides.in              manning.gr3             sflux
err.log                 outputs                 vgrid.in
hgrid.gr3               param.nml               windrot_geo2proj.gr3
hgrid.ll                run.log
launchSchism.sh         schism_model.json
```

```
!python3 -m json.tool schism/schism_model.json
```


```
{
    "global_grid": false,
    "geometry": {
        "lon_min": -30.0,
        "lon_max": -10.0,
        "lat_min": 60.0,
        "lat_max": 70.0
    },
    "lon_min": -30.0,
    "lon_max": -10.0,
    "lat_min": 60.0,
    "lat_max": 70.0,
    "coastlines": null,
    "start_date": "2017-10-01T00:00:00",
    "end_date": "2017-10-01T12:00:00",
    "date": "2017-10-01T00:00:00",
    "tag": "schism",
    "tide": false,
    "atm": true,
    "monitor": false
```

Rerun a case

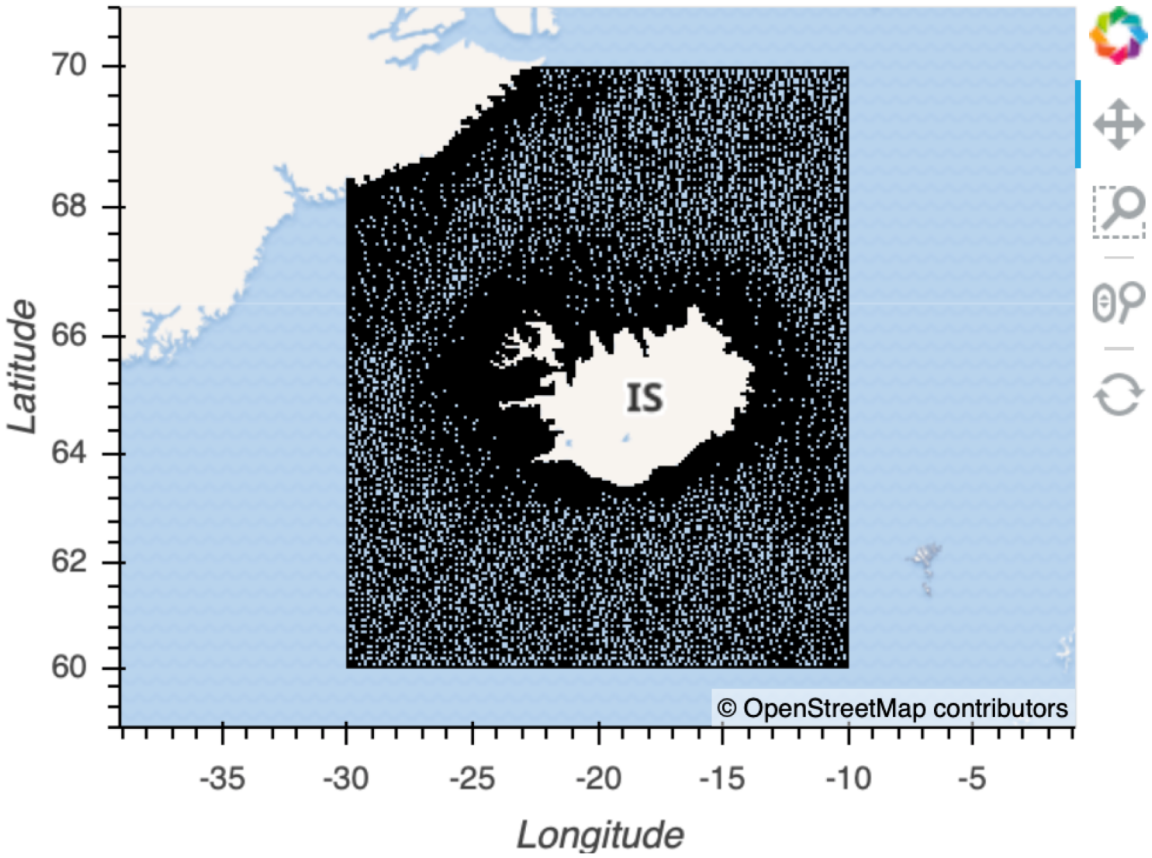
```
c = pyPoseidon.read_model('schism/schism_model.json')
c.execute()
```

Current Release : 0.4.0

- *with preliminary support for GMSH*
  - **More robust for HR coastlines**
  - **Many advanced features**

```python
import xarray as xr
grid = xr.open_dataset('ice.nc')

grid.hplot.grid(width=400, height=300)
```



European Commission

Utilities include

- *Adjust Bathymetry to coastlines*
- *Create a seam to a global grid for 2d visualizations*
- *Initiate and manage a forecast workflow*

Installation

```
conda install -c gbrey pyPoseidon
```

You can also install SCHISM with

```
conda install -c gbrey pschism
```

Current status of pre-operational setup

- Functional API for SCHISM in pyPoseidon (done)
- Conda integration (done)
- Work flow for operational storm surge forecasting (tested)
- Preliminary validation for Europe (completed)
- Integration of Waves (under testing)
- Inundation configuration (under testing)
- Global simulations (under testing)
- Tide integration (under testing)

Outlook - Milestones

- Reference Global grid
- Hindcast (ERA5)
- Validation/Verification/Skill
- Validated Integration of Tides/Waves
- Nesting
- Inundation
- Compound flooding
- pyPoseidon documentation

Long term goals

- Extend pyPoseidon to include new solvers and/or processes
- Encourage and support coastal topobathy analysis
- Support community of users and developers (hopefully)
- Produce (reproducible) datasets for usage in Disaster Risk Management
- Data analytics

Open issues

- Calibration of bottom friction coefficient
- Topobathy
- Satellite data (time acquisition frequency)
- 3D
- Validation data (in-situ data cleanup)

References

This presentation was made with RISE (https://rise.readthedocs.io/en/stable/index.html)

and is available on Github at

https://github.com/brey/PRESENTATIONS/blob/main/SCHISM_FEB_2021.ipynb