Norwegian Meteorological Institute

wavy - a package for wave model validation

From a CFOSAT user perspective

Patrik Bohlinger, Gaute Hope, Lars Robert Hole, Trygve Halse, Øyvind Breivik

12.09.2022

About me

- operational wave modelling WW3
- wave model validation
- statistical modelling
- non-stationary extreme values

Norwegian Meteorological Institute

README.md

wavy

conda test passing | Python build and test wheels passing | lint passing | coverage 43% | docs passing

Main developer and moderation:

Patrik Bohlinger, Norwegian Meteorological Institute, patrikb@met.no

Purpose

Package to aid the collocation of observations and wave model output as well as subsequent wave model validation. Observational sources can be satellite altimetry or in-situ data. A variety of open source alternatives is implemented.

Docs

For more information about the package as well as documentation please refer to the documentation.

Credits

When using **wavy** please give credit by citing: Bohlinger et al. 2019

Norwegian Meteorological Institute

Purpose of ***wavy***

- obtaining satellite data
- organizing observations for frequent use
-

Purpose of ***wavy***

- ❑ obtaining satellite data
- ❑ organizing observations for frequent use
- ❑ collocation with model output
- ❑ validation of collocated time series
- ❑

Purpose of ***wavy***

- ❑ obtaining satellite data
- ❑ organizing observations for frequent use
- ❑ collocation with model output
- ❑ validation of collocated time series
- ❑ get quick looks

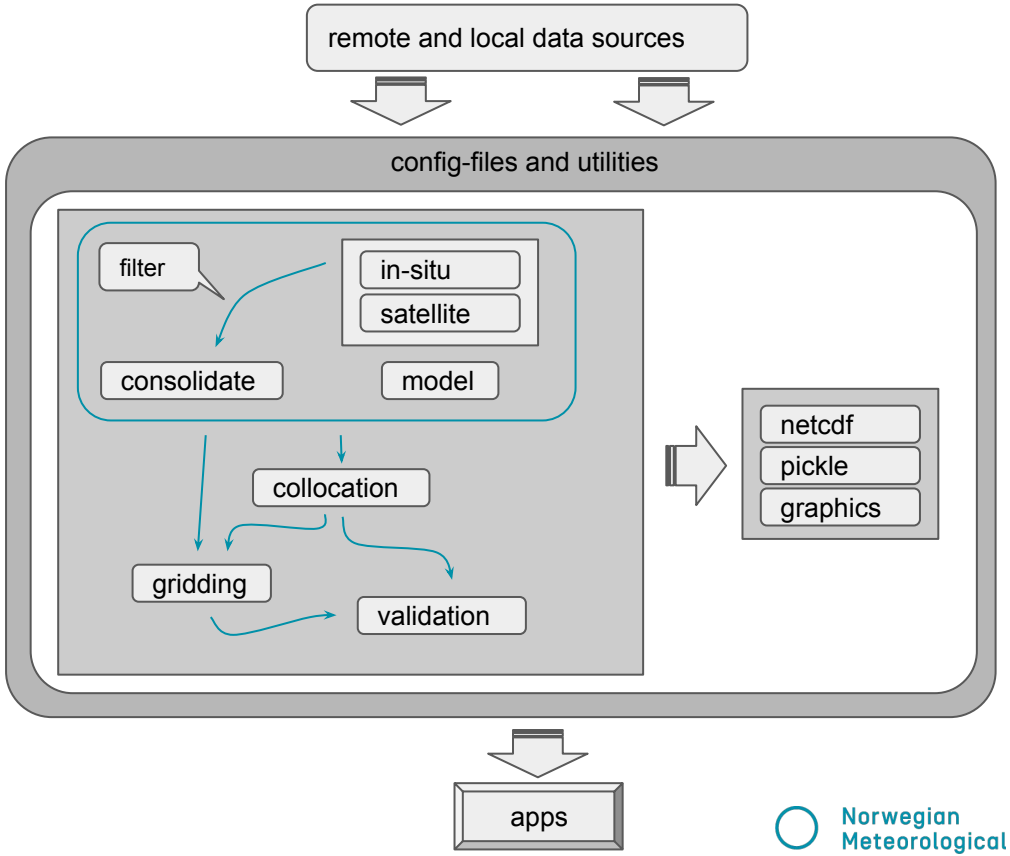
Purpose of ***wavy***

Ease collocation/validation
&
enable the comparison of different methods

Structure of wavy

Strategy:

- build on existing services
- use of open data
- use standards (e.g. cf)
- use config files to introduce flexibility
- tailored to marine applications
- for our purpose: **less is more**



Norwegian Meteorological Institute

Research

Near Real Time

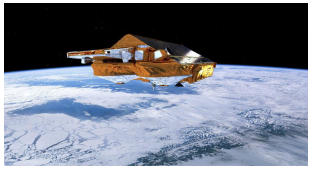
Climate Studies

Eumetsat L2

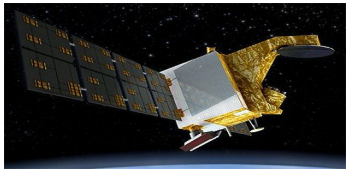
Sentinel-3A/B

Sentinel-6A MF

CryoSat-2



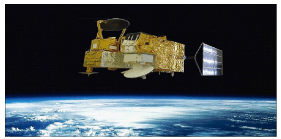
CFOSAT



CMEMS L3

Jason 3

HaiYang-2B



SARAL/AltiKa

CCI L2P

CCI L3

Jason 1,2,3

Envisat

ers-1, ers-2

GFO

TOPEX/Poseidon

CryoSat-2

SARAL/AltiKa

AVISO, etc ...

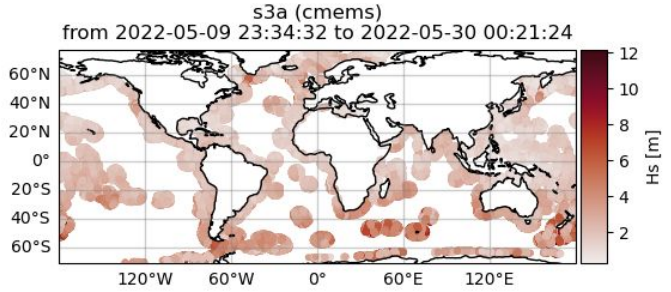
Norwegian Meteorological Institute

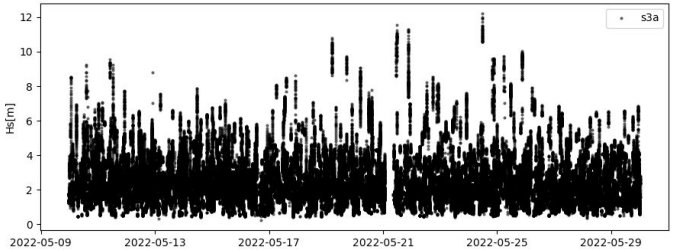
Example: altimetry

```
from wavy.satmod import satellite_class as sc

sco = sc( sdate      = "2022-5-20",
          edate      = "2022-5-25",
          region     = "global",
          filterData = True,
          land_mask  = True,
          dtc_mask   = True,
          dtc_llim   = 200,
          dtc_ulim   = 400 )

sco.quicklook(a=True)
```

Example: altimetry

```
from wavy.satmod import satellite_class as sc

sco = sc( sdate      = "2022-5-20",
          edate      = "2022-5-25",
          region     = "global",
          filterData = True,
          land_mask  = True,
          dtc_mask   = True,
          dtc_llim   = 200,
          dtc_ulim   = 400 )

sco.quicklook(a=True)
```

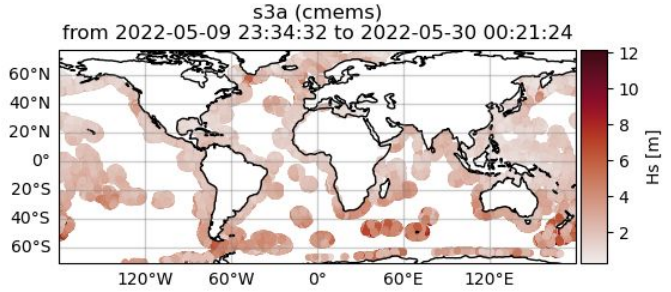


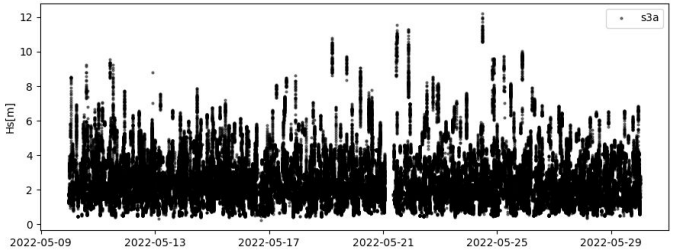


Norwegian Meteorological Institute

Example: altimetry

```
from wavy.satmod import satellite_class as sc

sco = sc( sdate      = "2022-5-20",
          edate      = "2022-5-25",
          region     = "global",
          filterData = True,
          land_mask  = True,
          dtc_mask   = True,
          dtc_llim   = 200,
          dtc_ulim   = 400 )

sco.quicklook(a=True)
```





Norwegian Meteorological Institute

Class object structure

satellite_class object:

- meta info
- class methods

```
>>> sco.
sco.edate                sco.quicklook(
sco.filter               sco.region
sco.filterSpecs          sco.sdate
sco.get_item_child(      sco.stdvarname
sco.get_item_parent(     sco.twin
sco.mission              sco.units
sco.obstype              sco.varalias
sco.path_local           sco.varname
sco.processing_level     sco.vars
sco.product              sco.write_to_nc(
sco.provider             sco.write_to_pickle(
```

- dictionary
- meta of source file

```
>>> sco.vars.keys()
dict_keys(['sea_surface_wave_significant_height', 'time',
'time_unit', 'latitude', 'longitude', 'datetime', 'meta'])
>>>
```


Norwegian Meteorological Institute

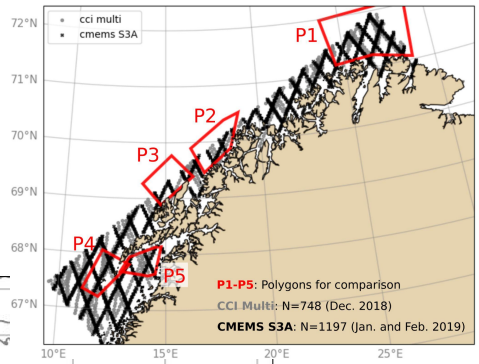
Regions

- regular lat/lon boundaries
-

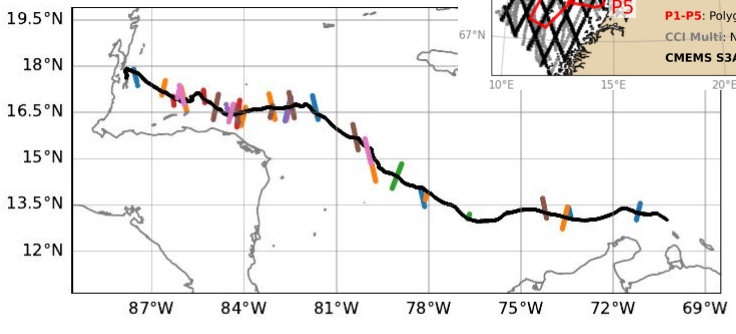
Regions

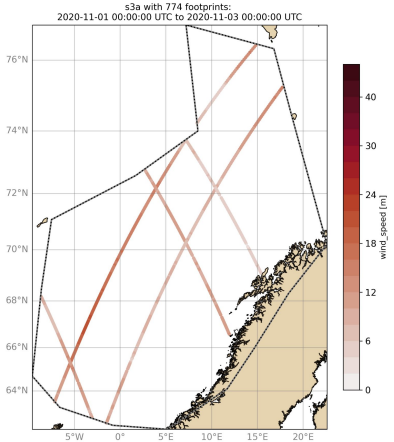
- regular lat/lon boundaries
- polygon
- geojson
- trajectory

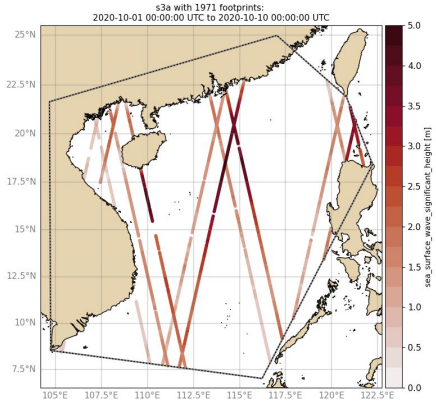
Halsne et al. 2022



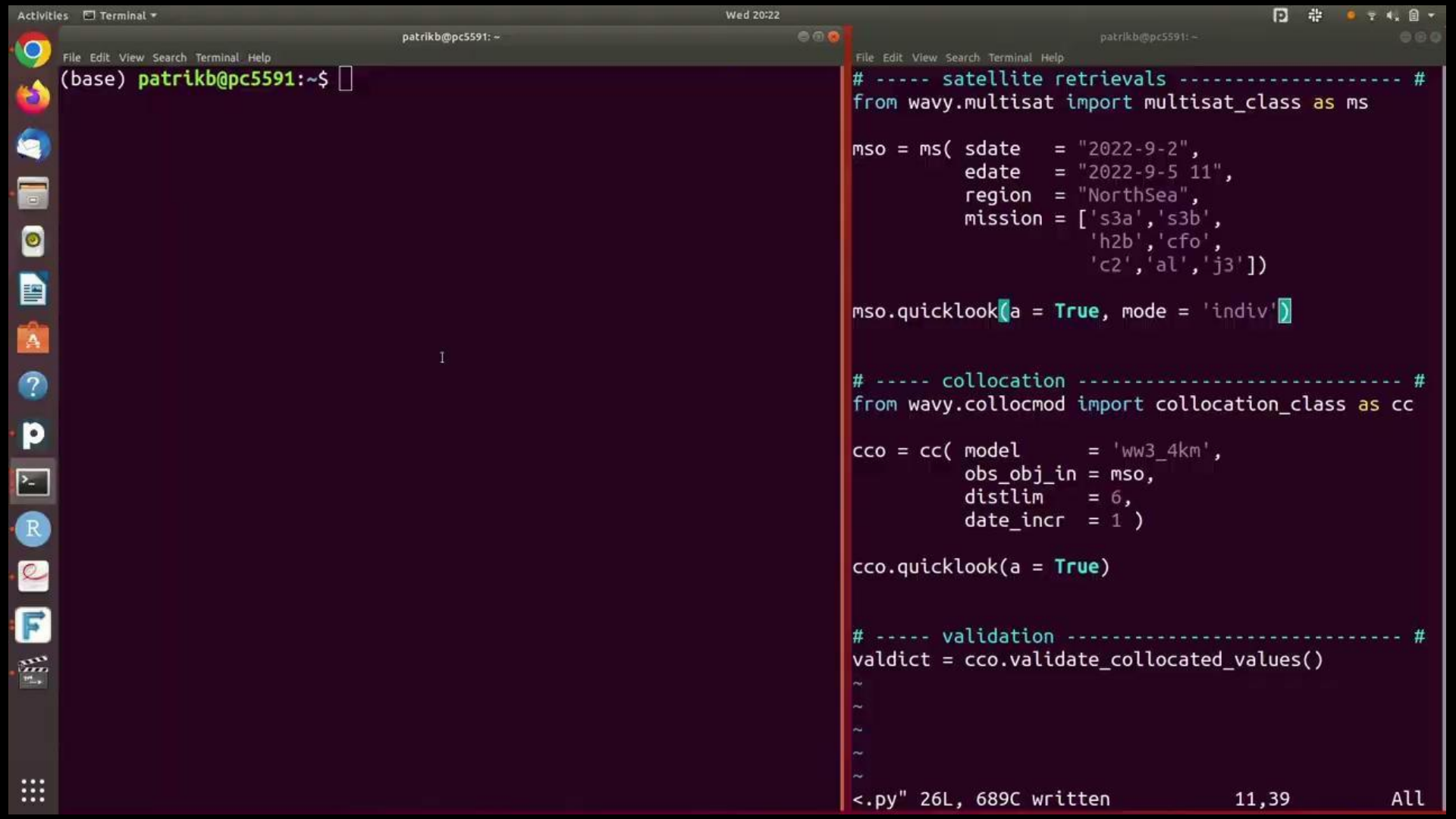
Rabault et al. 2022



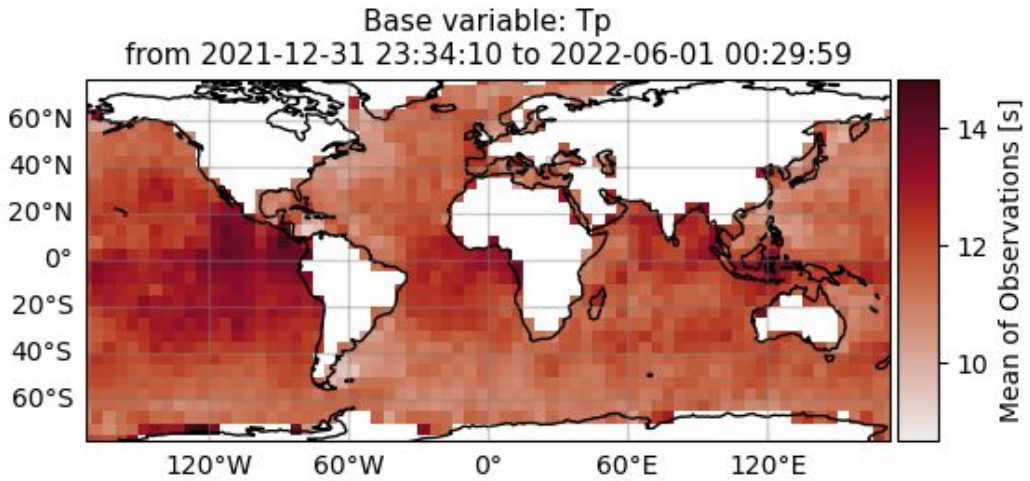




Example: Workflow


Activities
Terminal
Wed 20:22
patrikb@pc5591: ~
File Edit View Search Terminal Help
(base) patrikb@pc5591:~$
patrikb@pc5591: ~
File Edit View Search Terminal Help
# ----- satellite retrievals -------------------- #
from wavy.multisat import multisat_class as ms
mso = ms(sdate = "2022-9-2",
edate = "2022-9-5 11",
region = "NorthSea",
mission = ['s3a','s3b',
'h2b','cfo',
'c2','al','j3'])
mso.quicklook(a = True, mode = 'indiv')
# ----- collocation ----------------------------- #
from wavy.collocmod import collocation_class as cc
cco = cc(model = 'ww3_4km',
obs_obj_in = mso,
distlim = 6,
date_incr = 1)
cco.quicklook(a = True)
# ----- validation ------------------------------ #
valdict = cco.validate_collocated_values()
~
~
~
~
~
<.py" 26L, 689C written 11,39 All

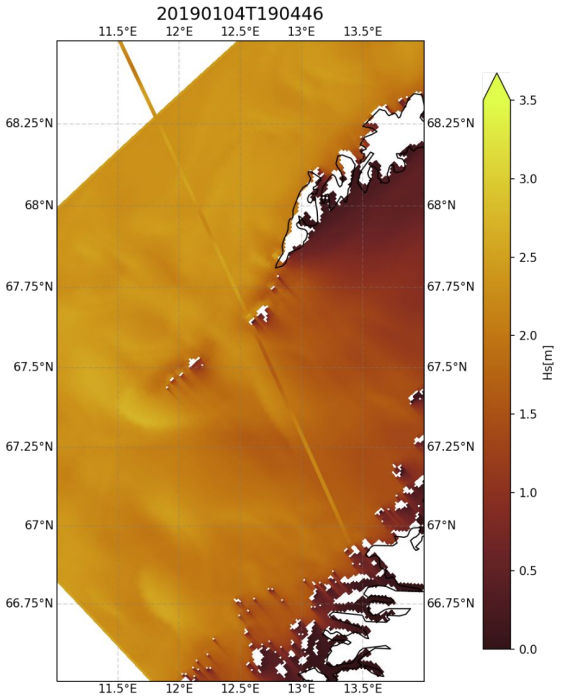
Example: CFOSAT (Tp gridded)

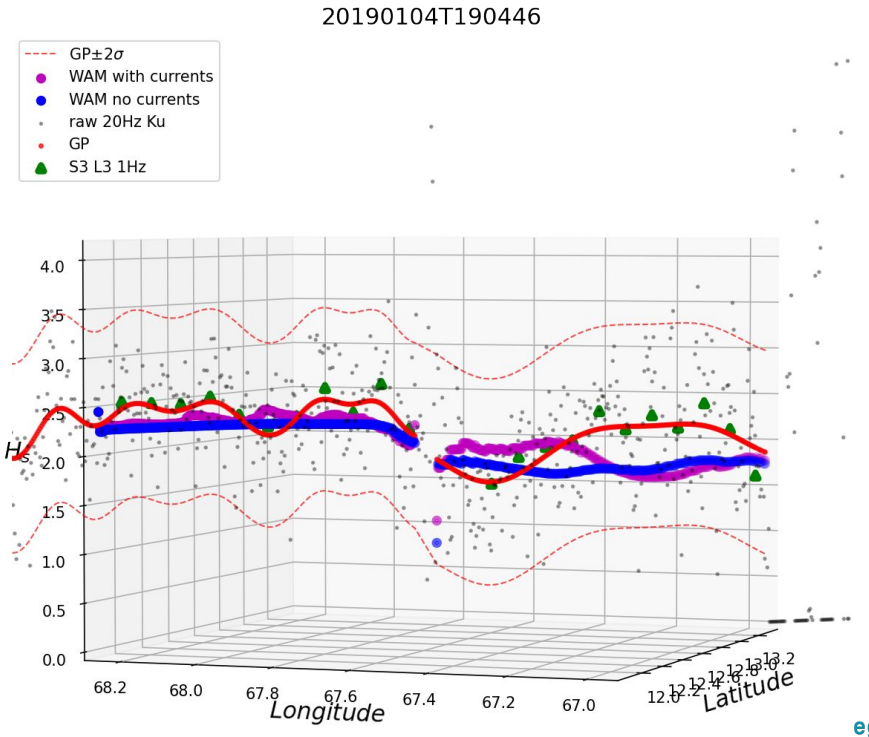



```python
# imports
from wavy.satmod import satellite_class as sc
from wavy.gridder import gridder_class as gc
from wavy.grid_stats import apply_metric

# retrieval
sco = sc( sdate = "2022-1-1",
          edate = "2022-6-1",
          mission = "cfo",
          product = "cfo_swim_L2P",
          varalias = 'Lp',
          return_var = 'Tp',
          region = 'global' )

# setup grid
bb = (-175,175,-80,80)
res = (5, 5)
gco = gc(oco=sco,bb=bb,res=res)

# ovals,mvals,Midx = gco.get_obs_grid_idx()
var_gridded_dict, lon_grid, lat_grid = \
        apply_metric(gco = gco,metric="all")

# quicklook
gco.quicklook( val_grid = var_gridded_dict,
               lon_grid = lon_grid,
               lat_grid = lat_grid,
               metric = 'mor')
```

Meteorological Institute

Example: Experimental





egian
orological
institute

Conclusion/Outlook for ***wavy***

- Practical validation tool for sea state parameters and wind
- Extension plans
  - variables like SLA or SST
  - filter methods for L2 (EMD, …)
  - spectral validation from CFOSAT
  - triple/multi collocation methods
- Collaboration