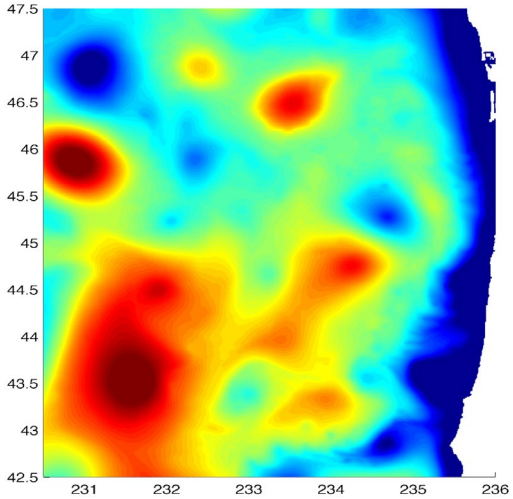
Ocean simulator for science applications

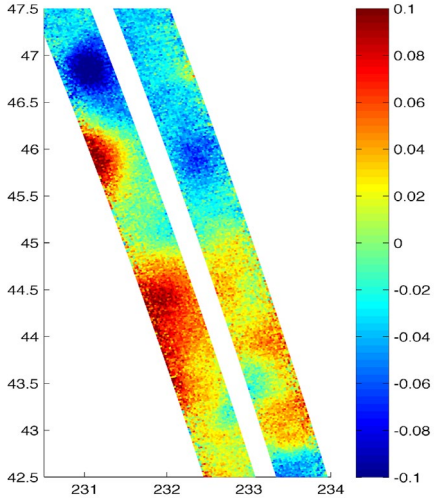
Clement Ubelmann, Lucile Gaultier, Lee-Lueng Fu, Jinbo Wang

SWOT simulator for Ocean Science

- The needs for a light/portable tool to easily simulate SWOT L2 data with realistic sampling and errors/noise has been pointed by the SDT team.
- **The tool relies on spectral error budget specifications from the project team**
  → It is NOT an instrument simulator.
- Open source, use git version or download tar on the github: https://github.com/SWOTsimulator/swotsimulator.git

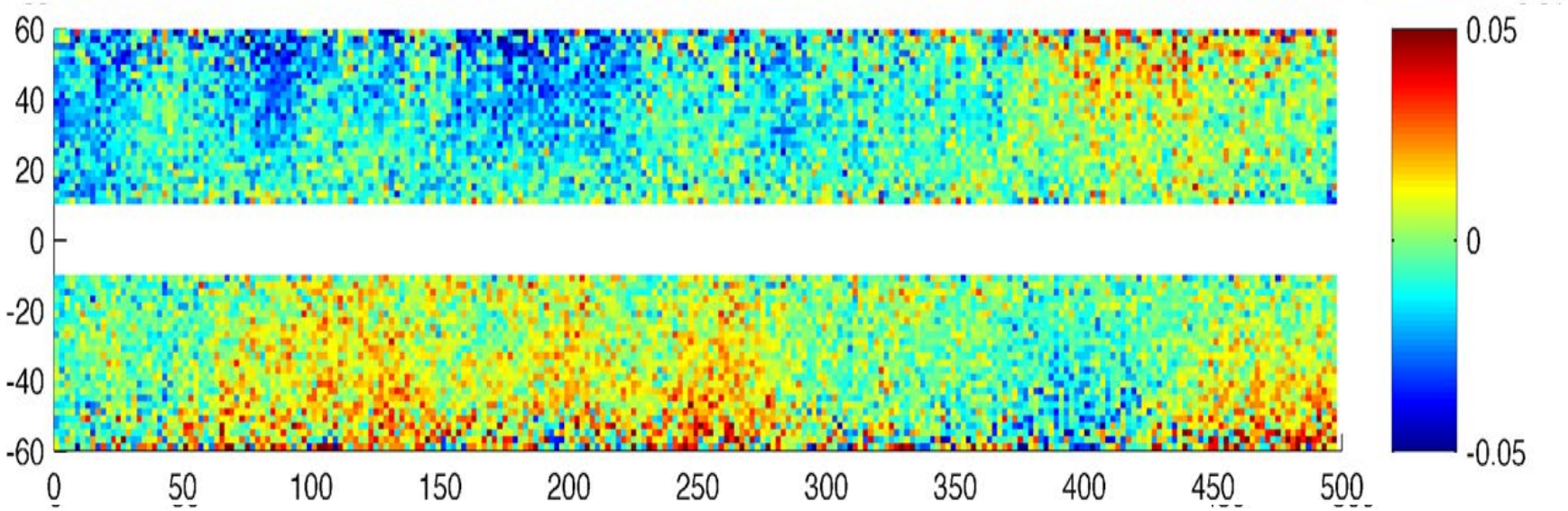
INPUTS: user's model Sea Surface Height Coordinates

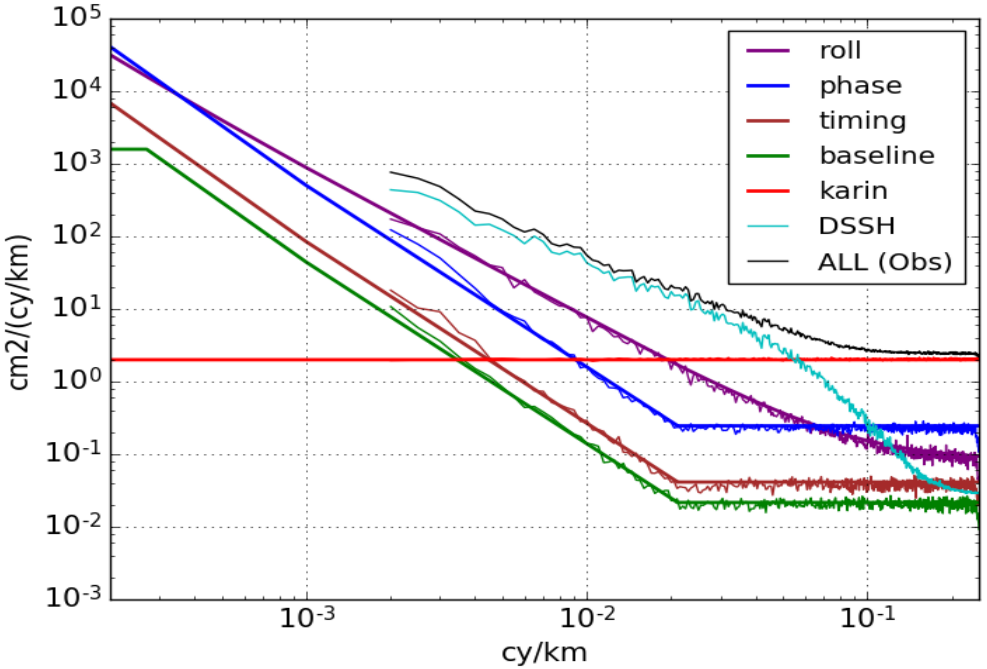




OUTPUTS:
SWOT synthetic data sampled on a swath grid
Each error is saved separately

Random error realizations





Playing with the simulator

Fill the parameter file:

```
# ----------------------#
# Files and directories
# ----------------------#
# ------ Directory that contains orbit file:
dir_setup='[yourpath]/SWOT_simulator/data/'
# ------ Directory that contains your own inputs:
indatadir='[yourpath_to_yourdata]/'
# ------ Directory that contains your outputs:
outdatadir='[yourpath_to_outputs]/'
# ------ Orbit file:
filesat=dir_setup+'/orbit292.txt'

# ----------------------#
# SWOT swath parameters
# ----------------------#
# ------ Satellite grid file root name:
#        (Final file name is root_name_[numberofpass].nc)
filesgrid=outdatadir+'/[your_grid_root_name}'
# ------ Force the computation of the satellite grid:
makesgrid=True or False
# ------ Give a subdomain if only part of the model is needed:
#        (modelbox=[lon_min, lon_max, lat_min, lat_max])
#        (If modelbox is None, the whole domain of the model is considered)
modelbox=None or [yourlon_min, yourlon_max, yourlat_min, yourlat_max]
# ------ Distance between the nadir and the end of the swath (in km):
halfswath=60.
# ------ Distance between the nadir and the beginning of the swath (in km):
halfgap=10.
# ------ Along track resolution (in km):
delta_al=1.
# ------ Across track resolution (in km):
```

A few technical possibilities

- Consider the provided orbits for the two phases of the mission:

Orbit	Repeat cycle (days)	Number of passes
Fast Sampling orbit	0.99349	28
Science orbit	20.8646	584

- Turn on long-wavelength roll errors : remove the 1,500 km cutoff that ‘simulate’ cross calibrations

- Possibility to simulate other altimetric observations (e.g. Jason, AltiKa, …): OSSEs with a constellation of nadir altimeters

Changelog for the last version

Thanks for sharing your feedbacks and issues.

- Bugs corrected: run altimeter simulator alone, shift time in orbits.
- Updates: Karin noise, science and one day orbit files
- Improve packaging of the code, readability of function, user proof and message error.
- Netcdf 3 format for input is not supported anymore
- Minimum configuration for python packages is required: python 3.3 or 2.6, numpy 1.5 …
- Plotting scripts for documentation and test scripts are on git.

Bottom line: ⚠ Correction of bugs thanks to issues reported
Simulator noise and orbits have been updated

Future improvement

Keep sharing your feedback and issues.

The new release at the end of the year will include:

- SWH varying with coordinates: climatology from WW3 model.
- Improve random noise sampling at long wave number
- Update swotsimulator output format to be as close as possible to the future SWOT L2 data format.
- Module for internal tides (See Ed's presentation)

Bottom line: Format of outputs are still evolving
Two ways to compute the noise.

Applications

- **The tool is designed to explore science applications:**

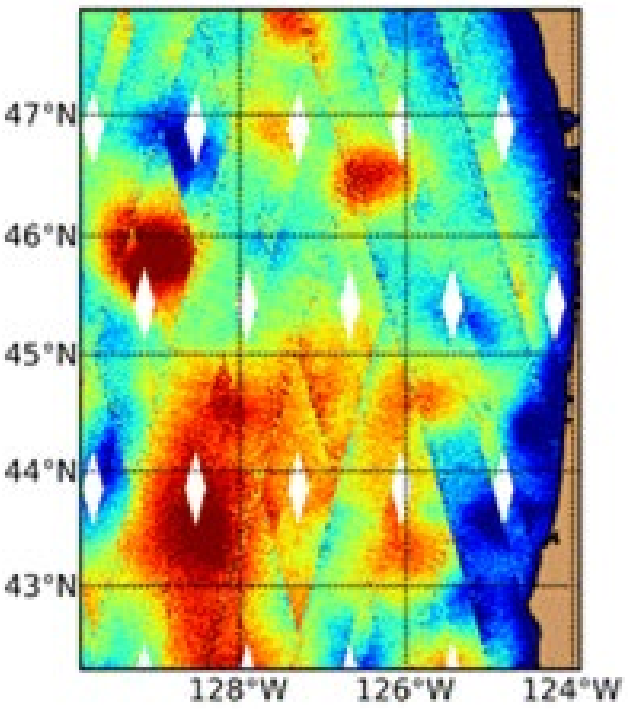
→ Consider the two mission phases: 1-day orbit and 21-day orbit: What science can we learn during the fast sampling phase

- Reconstruct the 2D signal: Deal with long time gaps to reconstruct continuous SSH, deal with high frequency waves and short mesoscales, Impact of the noise for high-order derivative quantities (e.g. vorticity)
- Compare the 2D signal with what other potential future mission: a similar simulator has been recently coded for the proposed SKIM surface current mission (ESA)

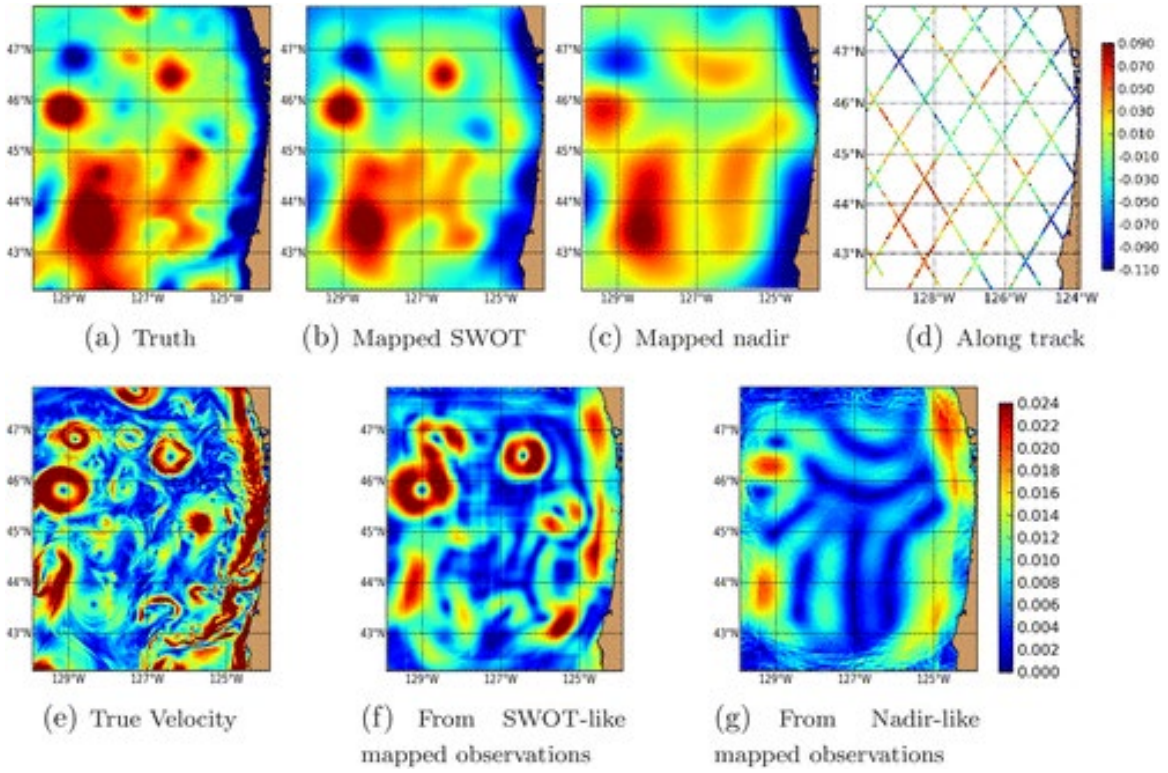
- **The tool can also be used to test calibration algorithms to improve L2 products:**

→ Roll error: test cross calibration techniques

The Challenge of Using Future SWOT Data for Oceanic Field Reconstruction (Gaultier&al., JTECH, 2016)



(c) SWOT-like SSH



(a) Truth (b) Mapped SWOT (c) Mapped nadir (d) Along track

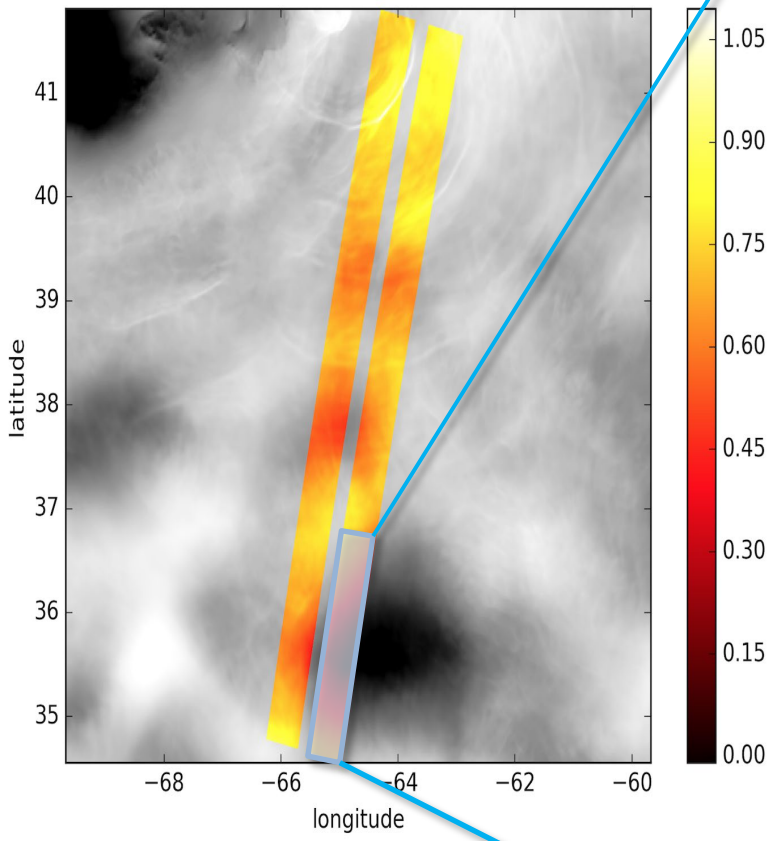
(e) True Velocity (f) From SWOT-like mapped observations (g) From Nadir-like mapped observations

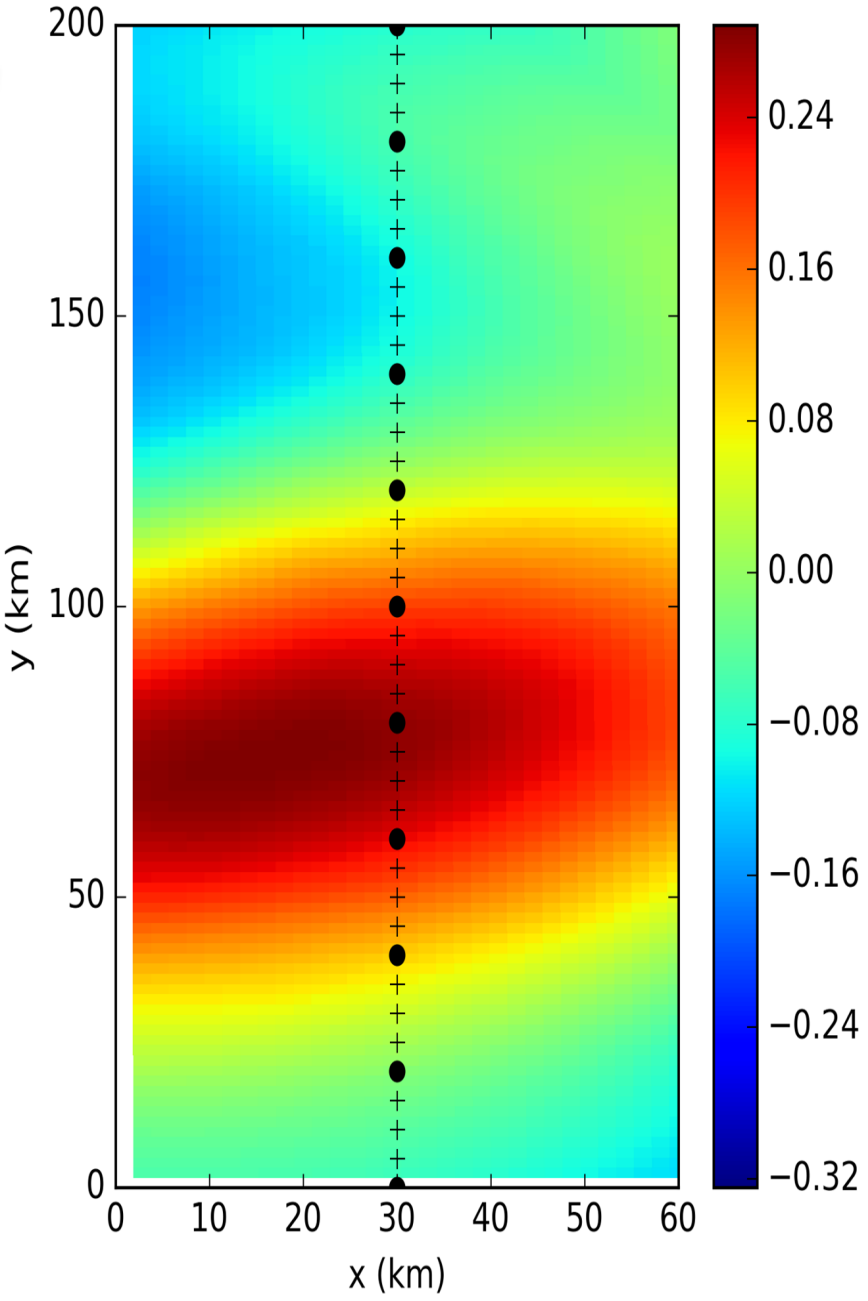
Evaluate how standard interpolation methods would work on SWOT data

Recent studies using the simulator

Building a realistic OSSE for CalVal: simulator run on the 1/48 MITgcm model run (Wang&Fu)

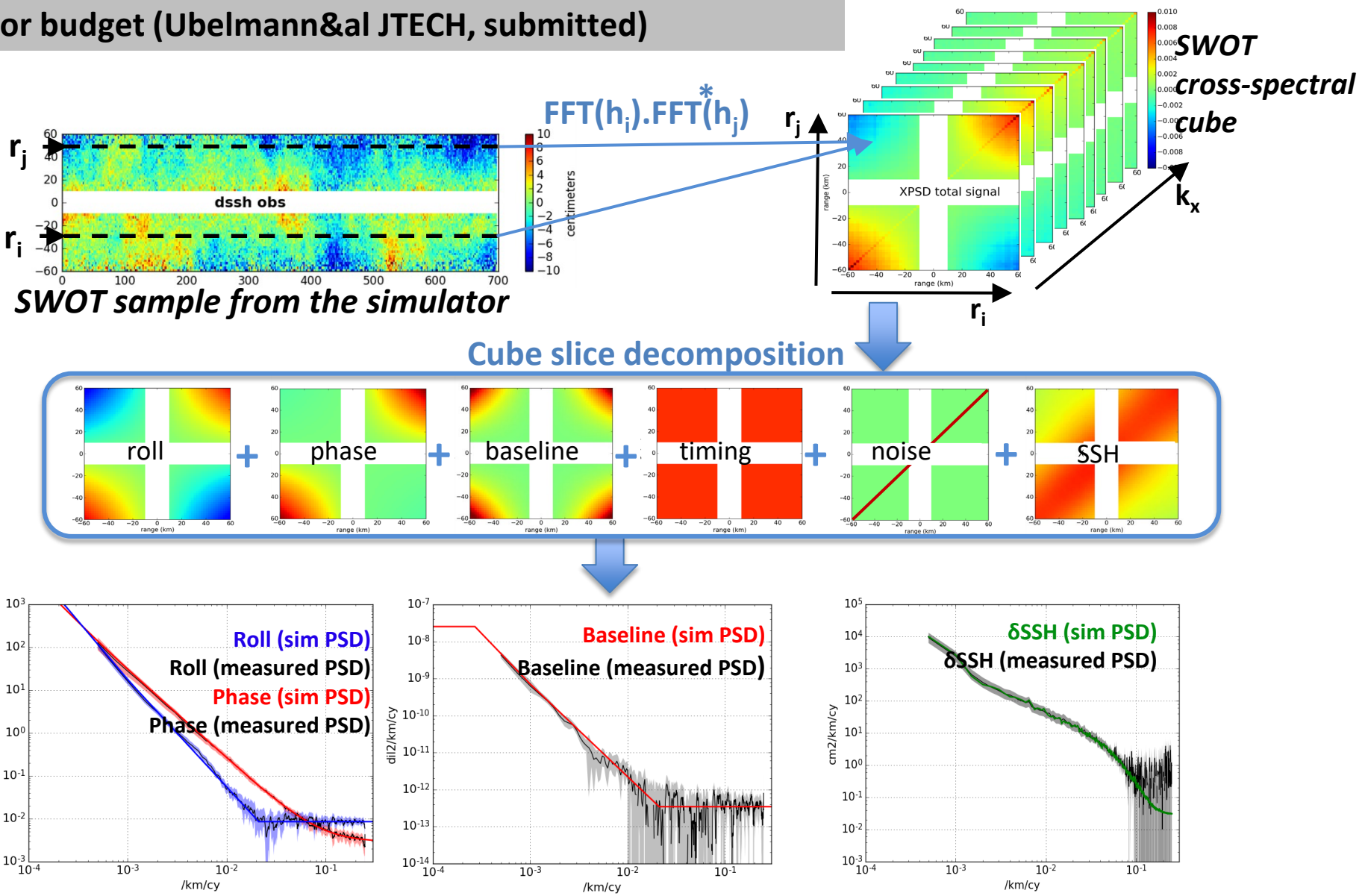
Simulator sample from LLC 4320





+ different in-situ scenarios including PIES, UCTD and moorings

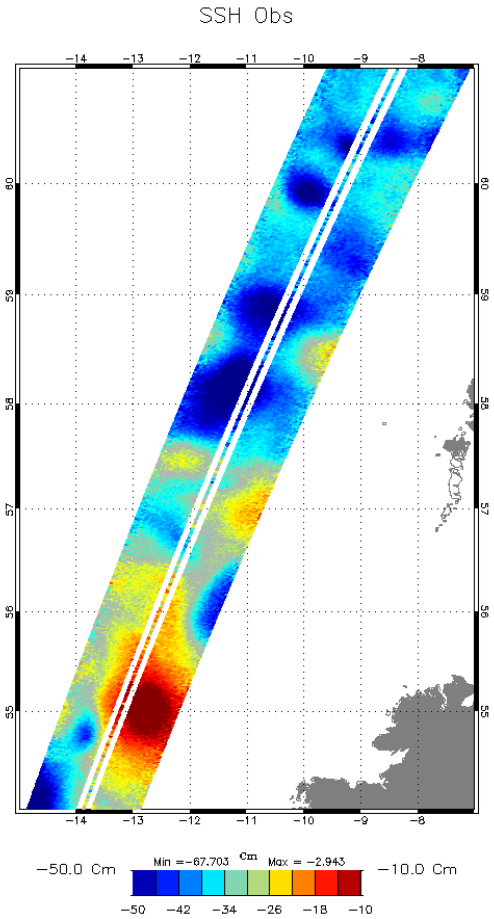
Test of a cross-spectral method to measure SWOT error budget (Ubelmann&al JTECH, submitted)



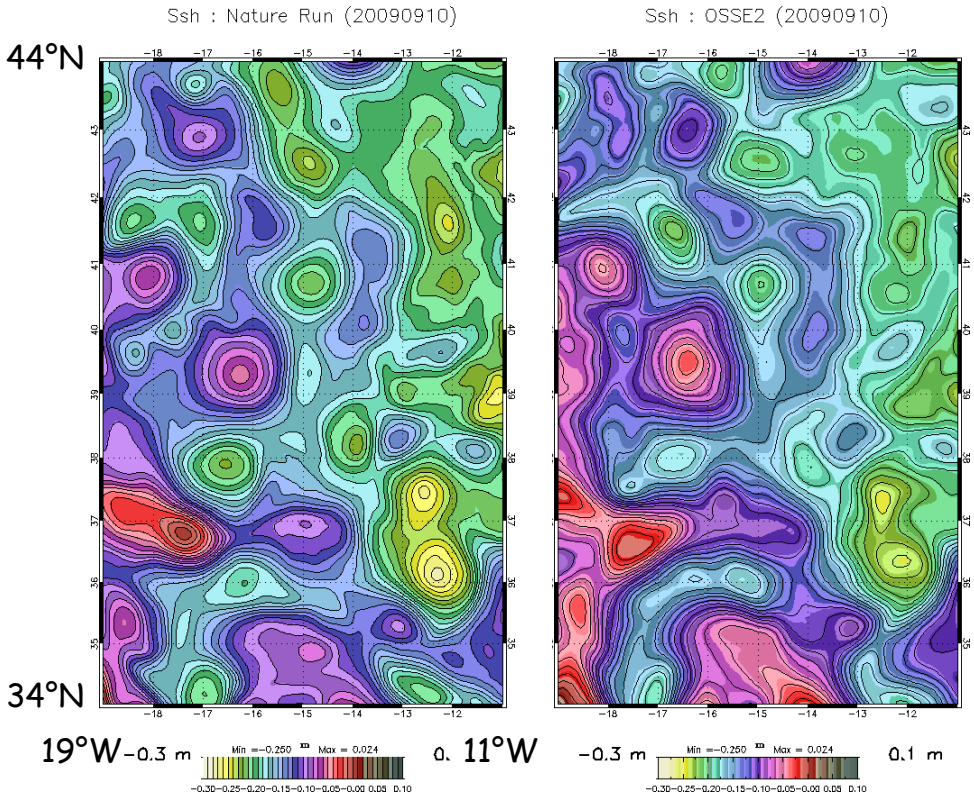
From the observed field, we can retrieve the simulated breakdown of SWOT error budget

Applications: CalVal and high-level products

Assimilation of SWOT altimetry data in Mercator Ocean system (Benkiran&Remy & Le Traon)



Simulated Karin+nadir altimeter data

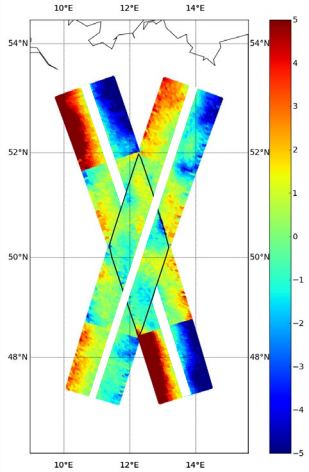


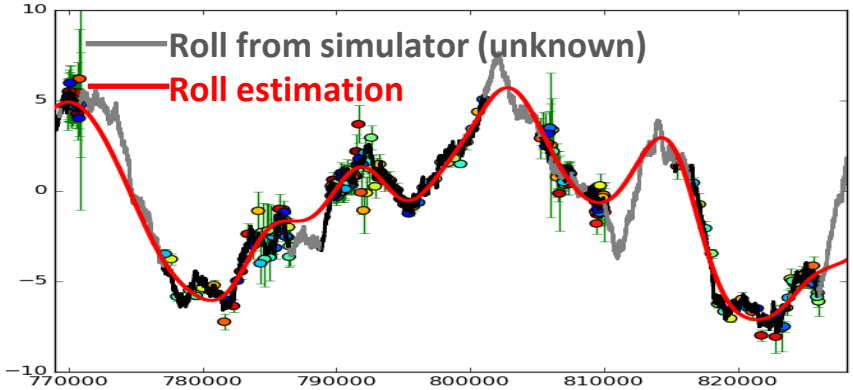
NR : Data
IBI36

OSSE2: Assim SWOT
(Forecast)

SWOT crossover calibration and inland propagation (Module to plug on Hydrology simulator, 2017)

Example of XOVER diamond in the simulator and local calibration





Residual error after Xover calibration

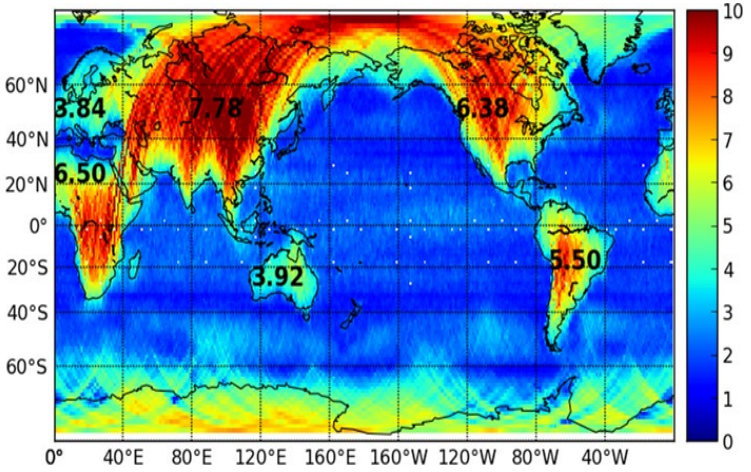
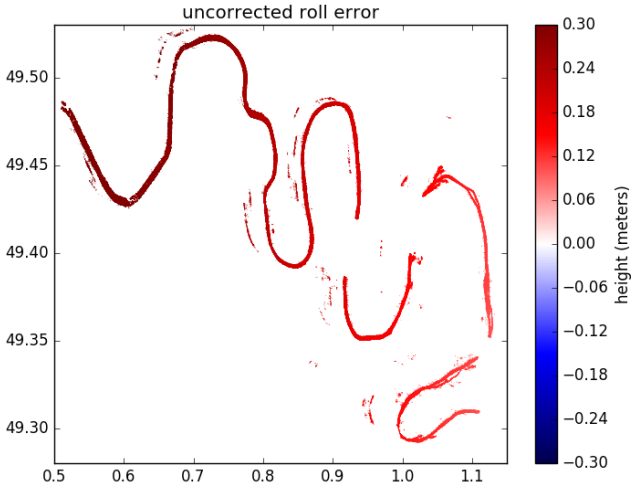
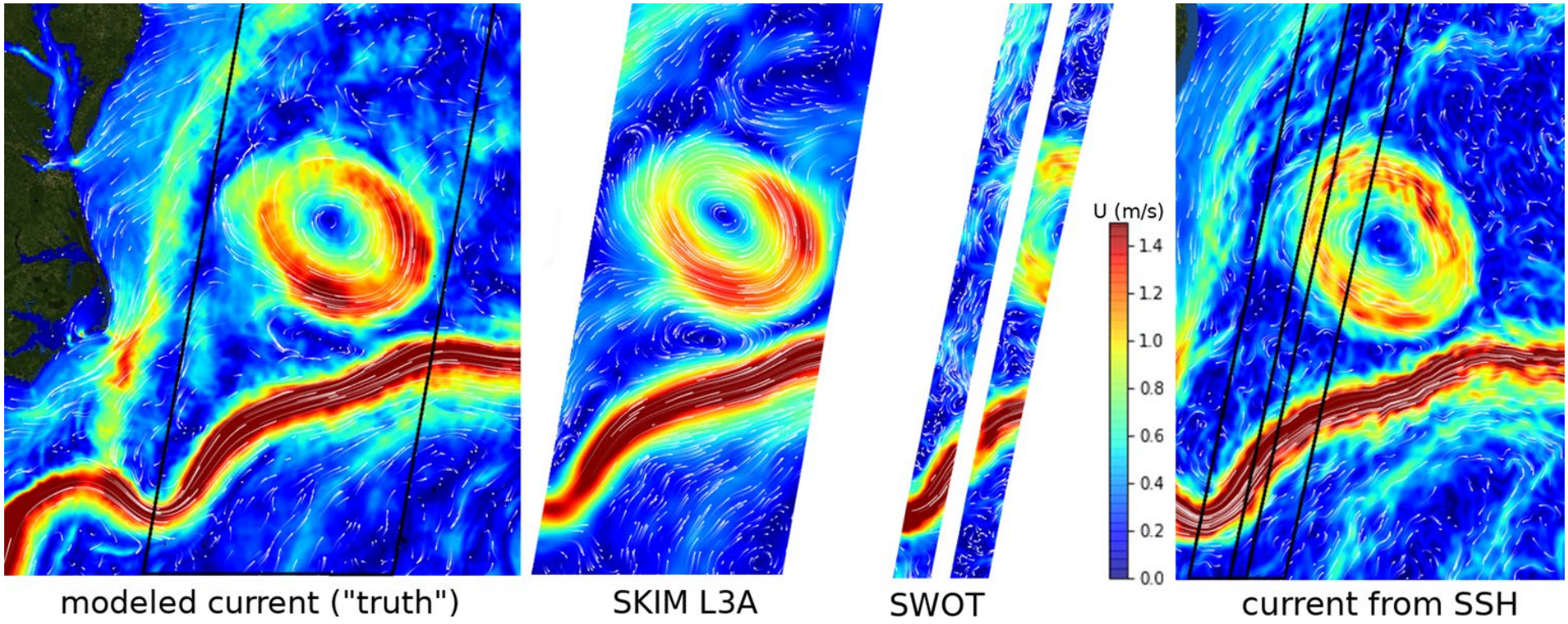


Illustration of roll error applied on the cloud point

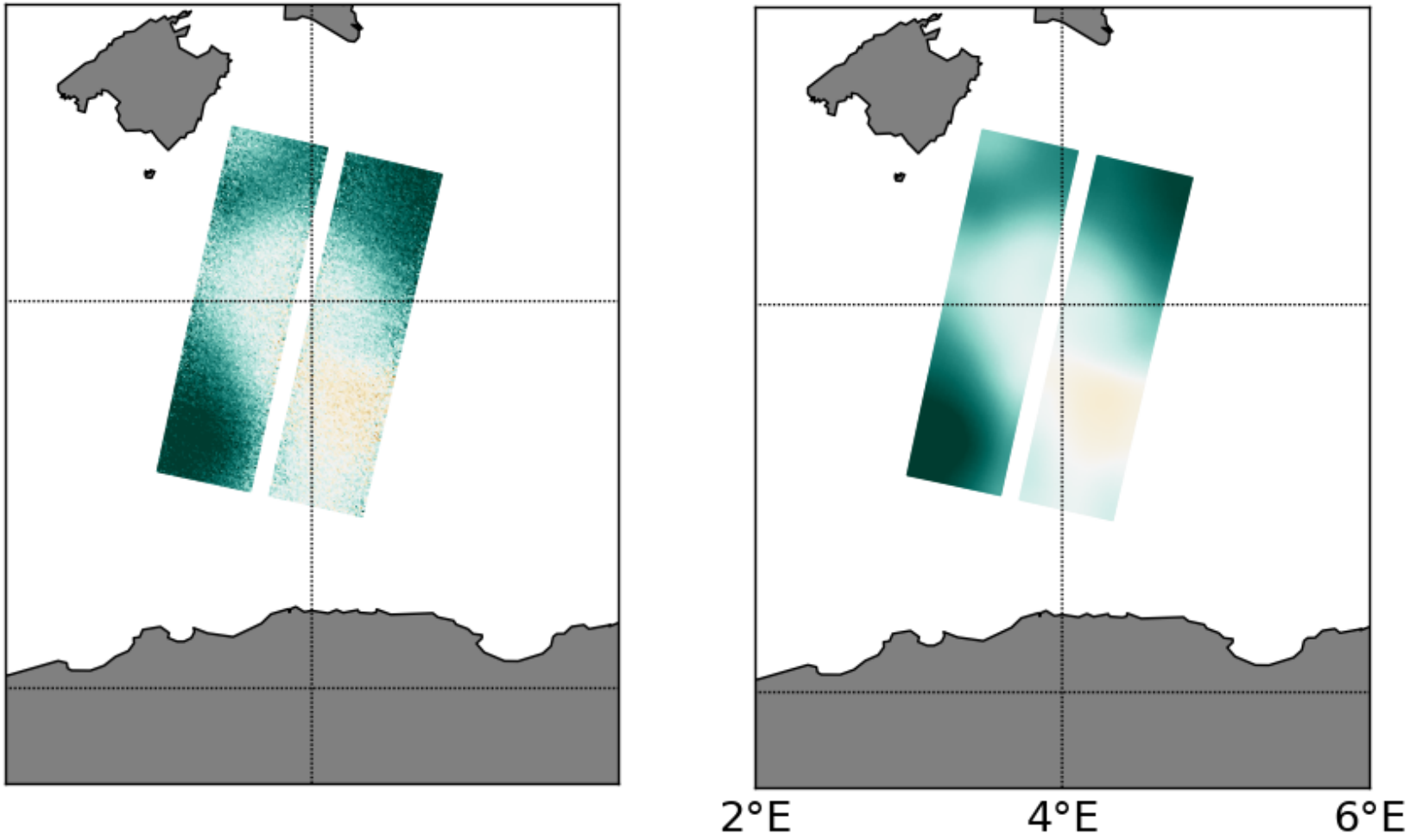


SKIM, *S*ea surface *KI*nematics *M*ultiscale monitoring, *full proposal for ESA EE9*
Ardhuin et al, 2017

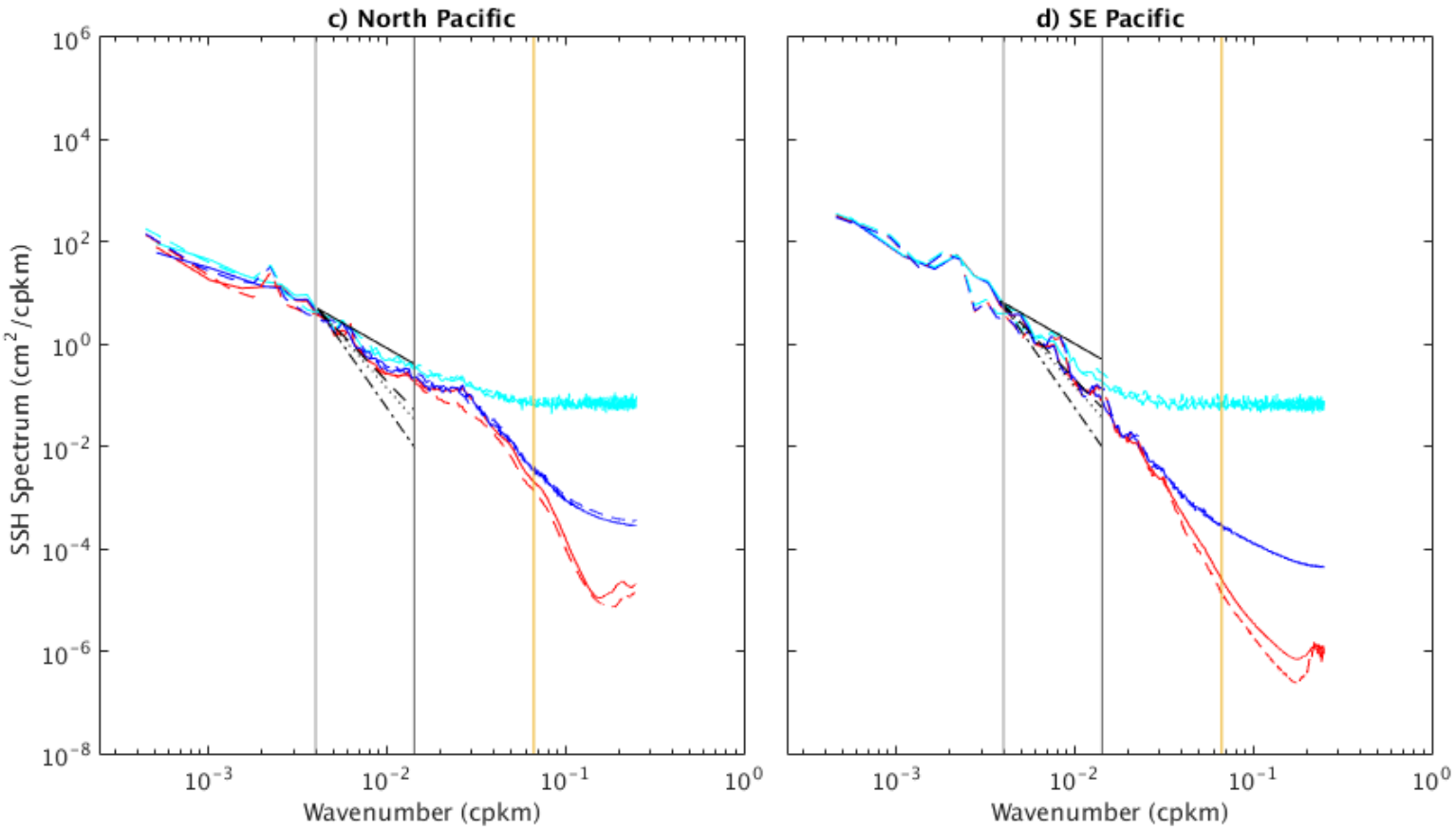


Compare currents retrieved from SWOT with the one from other potential future mission: SKIM

Testing advanced denoising techniques (Cosme&al.)



Explore spectrum reconstructions in key Oceanic regions (Ye&Savage&Arbic)



Studying the impact of temporal aliasing (blue) and SWOT instrumental errors (turquoise) on the restitution of the full spectrum (red)

How it will evolve ?

- ~20 active users so far
- Many inputs from users have been integrated in the current version. Thank you!

- Possible upcoming improvements:

  - SWH (x,y,t)
  - More realistic Karin noise (not just white)
  - Wet-tropo(x,y,t)
  - Simulated roll correction (not just a spectral cutoff)