Author's response to Lee Lueng Fu's Comments on 'On the resolutions of ocean altimetry maps'

We would like to thank Lee Lueng Fu for taking time to read our manuscript and for providing comments. We provide below the referee comments in red and our answer to each point in green.

This paper presents a comprehensive study of the resolution of a popular product of ocean altimetry maps issued by the DUACS (AVISO) system. The study used a spectral coherence method to determine the spatial and temporal resolution of the maps, using independent altimetry data (for spatial resolution) and tide gauge data (for temporal resolution). A host of issues have been addressed: the dependence on the number of altimeters, the evolution of the system over the altimetry era, the comparison with previous studies, the sensitivity to the signal/noise ratio. This paper is highly recommended for users of the DUACS maps for any quantitative studies.

Some minor technical comments:

Lee Lueng Fu (**LLF**): P.1 lines 10-13: The sentence "These maps are ..." is awkward. Needs re-phrasing.

Author's Response (**AR**): we propose to re-phrase the sentence as: "These maps are generated using an optimal interpolation method applied to altimeter observations. They are provided on a global ¼°x ¼° (longitude x latitude) and daily grid resolution framework (1/8°x1/8° longitude x latitude grid for the regional products) through the Copernicus Marine Environment Monitoring Service (CMEMS)"

LLF: P.2, Line 30: What is the source for the independent 1 Hz along-track data? Although the answer is provided later, it should be made clear here.

AR: we add the source "Level 3 CMEMS" for the independent 1Hz along-track data

LLF: P.3, lines 25-30: Why does MSLA contain no noise? Is noise the same as random error? What is t in equations1-2? what is v(t)?

AR: We assumed that the MSLA contains no noise since it is resulting from the filtering of the along-track data through the DUACS mapping. We considered that the noise level in along-track SSH is much larger than in the mapped SSH. Here noise refer to instrumental noise. Following the comment made by referee #3, we have clarified this, and in the revised version of the manuscript, we redefined the effective resolution based on the spectral Noise-to-Signal ratio (PSD mapping error / PSD_along_track), which is more robust than the spectral coherence. This ratio (PSD mapping error / PSD_along_track) checks both the phase and the amplitude consistency between the two signals considered, whereas the spectral coherence focuses only on the phase consistency. We also illustrate that this change of definition has a weak impact on the results since the signal amplitude is globally optimal at the wavelength where the phase becomes incoherent. In response to reviewer #3, several test cases based on Observing System Simulation Experiment have been conducted and are freely available and interactively repeatable here: https://mybinder.org/v2/gh/mballaro/notebook.git/master (under the analysis_OSSE_NATL60 folder). They are also available at the end of the response to referee #3. These test cases show that the Magnitude squared coherence and Ratio PSD

error/PSD along-track are in good agreements, meaning the phase consistency is the dominant factor controlling the "quality" of the DUACS maps.

The ‘t’ variable corresponds to the time.

LLF: P.4, equation (6): where is Gvv from equation (4)? P.5, lines 17-18: Provide reference on the larger correlation scales.

AR: We used similar notation as in Bendat et al. (2010). As for the previous question, we have updated the mathematical derivation in the new version of the manuscript. The reference (Pujol et al., 2016) on the DUACS-DT2014 reprocessing is added as reference for the larger correlation scales.

LLF: P.7, line 5: add "than" after"smaller".

AR: “Than” will be added after “smaller”

LLF: P.7, lines 26-27: Elaborate why up to four altimeters are required for near-real time products.

AR: We mention why up to four altimeters are required for near-real time products, referencing to the Pascual et al. (2006) paper. Up to four altimeters are required for near-real time products because only passed observation are available for the mapping. This reduced number of observations has an impact on the estimation of the sea surface height.

LLF: P.7, lines 29-30: This is an important point worth noting when the use of independent track for the study is first mentioned in the paper.

AR: We also add a sentence about this point in the methodology section. “It is worth noting, that we probably underestimate the resolution capability of the maps since we are estimating the spatial effective resolution of degraded maps to keep an independent dataset aside.”

LLF: P.12, equation(9): either show a figure of SR(lambda) or describe its variation with lambda in words.

AR: We add a figure showing the SR(lambda), as well as for other spectral approach used to estimate the resolution (See Figure A5).

Author's response to Anonymous Referee #2 Comments on 'On the resolutions of ocean altimetry maps'

Anonymous Referee (AR#2): This paper: "On the resolutions of ocean altimetry maps" uses spectral coherence between SSH maps and along-track and tide gauge SSH measurements. The calculations seem correct and the figures are interesting, but to me, this approach is complicated because it combines together the resolution of the maps and the lengthscales of the processes being imaged. For example, in the equatorial region they estimate spatial resolution at 800km, far bigger than the altimeter track spacing.

Chelton, a co-author on this paper, has done a lot of work on evaluating sampling from satellite observations of both SSH and wind (Schlax, et al., 2001), and has examined the transfer function of linear mappings (Schlax and Chelton 1992). The latter paper focuses on the effects of the mapping as a smoother of the original field, examining the resolution of the mapping independently from the length scales of the mapped fields. A similar approach could be used here, taking out the correlation of the underlying fields to focus on the smoothing done by the mapping.

This is what I expected from the analysis, and I feel that the differences should be discussed and the results presented here put in context as a combination of two effects.

I would also like to request that the authors please also address why they do not compute decorrelations in physical space and time instead of coherence. This would allow the preservation of spatial structure that is removed by the stationarity assumption built into the coherence calculation, including the averaging over large regions to get adequate coherence statistics. Since only a decorrelation distance is reported, the sacrifices needed to be able to make the coherence calculation do not seem necessary, and a region-by-region decorrelation scale could have been reported. This would still have mostly represented the scale of the SSH field, not the mapping, but it would be simpler to compute, report, and understand.

references:

Sampling errors in wind fields constructed from single and tandem scatterometerdatasets. J. Atmos. Oceanic Tech., 18, 1014-1036, 2001. (Schlax, M. G., D. B. Chel-ton, and M. H. Freilich.)

Schlax, M. G. and D. B. Chelton, 1992: Frequency domain diagnostics for linearsmoothers. J. Amer. Stat. Assoc., 87, 1070{1081.

Author's Response:

We would like to thank the Anonymous Referee #2 for taking time to read our manuscript and for providing comments. The referee pointed out 2 main concerns:

- An approach based on transfer function (Schlax and Chelton 1992) should have been done and discussed
- A calculation in physical space should have been undertaken (a region-by-region decorrelation scale could have been reported)

Following the study by Dufau et al. (2016), who estimated the along-track resolution using spectral approach, we have made the choice to pursue the investigation of the maps resolution also using spectral approach. We agree with the reviewer's comment stating that an approach based on the estimation of the transfer function could have been undertaken and discussed. Hence, we propose to extent our analysis and compare our estimation of the effective resolution with the methods based on the transfer function for the estimation of the resolution capabilities. A paragraph is added in the Appendix A to discuss about the difference between the two approaches, as well as other approach such as the spectral magnitude ratio.

More importantly, following the comment made by referee #3, we redefined the effective resolution based on the Noise to Signal spectral ratio (PSD mapping error / PSD_along_track), which is more robust than the spectral coherence. This ratio (PSD mapping error / PSD_along_track) verifies both the phase and the amplitude consistency between the two signals considered, whereas the spectral coherence focuses only on the phase consistency. The main conclusions of the paper are unchanged. We also illustrate that this change of definition has a weak impact on the results since the signal amplitude is globally optimal at the wavelength where the phase becomes incoherent. In response to reviewer #3, several test cases based on Observing System Simulation Experiment have been conducted and are freely available and interactively repeatable here: https://mybinder.org/v2/gh/mballaro/notebook.git/master (under the analysis_OSSE_NATL60 folder). They are also available at the end of the response to referee #3. These test cases show that the Magnitude squared coherence and Ratio PSD error/PSD along-track are in good agreements, meaning the phase consistency is the dominant factor controlling the "quality" of the DUACS maps.

The spectral coherence estimates the correlation for each specific wavelength. The correlation in physical space is hence the integral of the spectral coherence. Computing the correlation in physical space would hence give strong value since the large-scale signal is globally well mapped in the DUACS system.

Author's response to J. Thomas Farrar's Comments on 'On the resolutions of ocean altimetry maps'

First of all, we would like to thank Tom Farrar for taking time to read our manuscript and for his detailed and inspiring review.

We attach below the entire review made on April 2, 2019. The main concern of the review is the fact the coherence-based measure is not well adapted for estimating the resolution. The review shows several 1D general examples of the behavior of various proposed measures of the resolution (e.g., magnitude squared coherence, filter transfer function), considering low, high noise level and noise free in the input signal. For the 1D example without noise, the squared coherence and filter transfer function highlight different behavior. Similar conclusions are found for the 1D example with low noise level. In a more realistic example with high noise level, the squared coherence gives information about the input SNR but nothing about the filtering.

We agree that the filtering method for similar cutoff wavenumber can impact the coherence amplitude. This impact only happens below the cutoff frequency (Figure 1 in the review) where we should not have any coherence with independent altimetry profiles. If we have some, this means that the DUACS filter has been too aggressive and we may afford shorter correlation scales. In that case, the coherence could be misleading. To avoid that, we can consider evaluating the ratio PSD(SSH_altrack - SSH_map)/PSD(SSH_altrack), i.e., noise-to-signal ratio.

It is important to mention that our computation of the Magnitude Squared Coherence takes only into account the phase consistency between the SSH_altrack signal and the SSH_map signal (not the amplitude). For example, if the correlation scales in DUACS were larger (resulting in smoother SSH, attenuated amplitude), we could have similar phase scores but a poor PSD ratio score. We made the test, artificially smoothing the maps: we obtain the same coherence score but a low PSD ratio score. This convinced us that the coherence is indeed not sufficient as pointed out in the review, but the PSD(SSH_altrack - SSH_map)/PSD(SSH_altrack) score should well characterize the skills, by penalizing both amplitude and phase.

Hence, we propose to switch to the definition of the resolution limit of the maps as **PSD(mapping_error)/PSD(SSH_true) = 0.5** (i.e., the wavelength where the mapping error are two times smaller than the true SSH signal).

In order to illustrate the assessment of the resolution based this new definition, we performed analysis on Observing System Simulation Experiment (OSSE). The details of the simulation and methods are presented in notebooks below; and are freely available / interactively repeatable here: https://mybinder.org/v2/gh/mballaro/notebook.git/master (under the analysis_OSSE_NATL60 folder).

We have performed 3 study cases:

- STUDY CASE #1: Comparison with independent along-track (similar to the approach used in the manuscript)

- STUDY CASE #2: Comparison with NON-independent along-track

- STUDY CASE #3: Comparison with independent along-track with higher instrumental noise that mapping error

The advantage of using OSSE is that we have access to all quantity we want:

- PSD(SSH_map)

- PSD(SSH_true)

- PSD(mapping_error)

- PSD(instrumental_error)

- PSD(SSH_obs) = PSD(SSH_true + instrumental_error)

With real DUACS dataset we can only get:

- PSD(SSH_map)

- PSD(SSH_obs)

- PSD(instrumental_error)

We show in the notebooks that we can approximate the ratio PSD(mapping_error)/PSD(SSH_true) with the three quantities (PSD(SSH_map), PSD(noise_SSH), and PSD(instrumental_error)).

PSD(mapping_error)/PSD(SSH_true) = [PSD(SSH_obs - SSH_map) - PSD(instrumental_error)] / [PSD(SSH_obs) - PSD(instrumental_error)]

The main conclusions from the STUDY CASE #1 notebook are:

- the resolution estimated with the magnitude squared coherence is in good agreement with the PSD(mapping_error)/PSD(SSH_true) = 0.5 approach, linked to the fact that the signal amplitude is globally optimal at the wavelength where the phase becomes incoherent, and thus the major concern for the DUACS system is in the phase consistency between SSH_altrack and SSH_map signal rather than in their amplitude.

- the ratio **PSD(SSH_obs - SSH_map)/PSD(SSH_obs) (blue curve)** and **[PSD(SSH_obs - SSH_map) - PSD(instrumental_error)] / [PSD(SSH_obs) - PSD(instrumental_error)]** (yellow curve) are similar to the **PSD(mapping_error)/PSD(SSH_true) (red curve)** for wavelength > 70km

- the ratio **PSD(SSH_obs - SSH_map)/PSD(SSH_obs) (blue curve) or [PSD(SSH_obs - SSH_map) - PSD(instrumental_error)] / [PSD(SSH_obs) - PSD(instrumental_error)]** (yellow curve) can thus be used to estimate map resolution

- the signals for wavelength < 50 km (grey area) are not meaningful since we are under the grid spacing of the DUACS gridded product as well as in the instrumental noise level

- the sensitivity of the ratio PSD(SSH_obs - SSH_map)/PSD(SSH_obs) to the PSD(instrumental_noise) is weak for wavelength > 70km

Conclusions in STUDY CASE #1 are also valid for STUDY CASE #2.

For STUDY CASE #3 with unrealistic high instrumental noise, the ratio PSD(SSH_obs - SSH_map)/PSD(SSH_obs) becomes extremely sensitive to the instrumental noise level.

Note that this STUDY CASE #3 is not happening in DUACS processing.

In conclusion, we have made the choice to change our definition of the measure of the resolution in the revised version of the manuscript. It was previously based on the Magnitude squared coherence. It is now based on the Noise-to-Signal ratio, which is more robust penalizing both amplitude and phase consistency between the two signals. We illustrate using OSSE that the definition of the resolution based on Noise-to-Signal ratio or the Magnitude squared coherence are equivalent for the DUACS system. The main outcomes of the paper are thus unchanged. We obtain similar spatial and temporal resolutions since the signal amplitude in DUACS is globally optimal at the wavelength where the phase becomes incoherent.

We have thus performed the analysis of the DUACS maps using the NSR ratio measure. The figures have been updated in the revised version of the manuscript. A comparison of various approach to estimate the resolution capability is added and provided in the Appendix A of the paper. It includes spectral magnitude ratio approach and the transfer function approach.

Example_OSSE_obs_NOT_mapping_NATL60

April 18, 2019

0.1 CASE STUDY 1: COMPARISON WITH INDEPENDENT ALONG-TRACK

INTRODUCTION

We here propose to show the behaviour of the spectral analysis in an OSSE case study based on NATL60 simulation. The input comes from hourly SSH data from a 1 year long period of the run NATL60-CJM165 made at IGE (Grenoble). We used the SWOTsimulator to sample alongtrack data (nadir-like dataset) on this simulation. The constellation is similar to the 20030101-20031231 period: Jason 1, Envisat, Geosat2, Topex/Poseidon interleaved. SWOTsimulator gives us the true SSH (SSH_{model}) and a pseudo-obs SSH (SSH_{obs} = SSH_{model} + white noise). These pseudo-obs SSH_{obs} are then used in the DUACS mapping. This case study 1 is similar to case study 2 except that we are doing here the evaluation of the maps against indepedent along-track data.

CASE STUDY #1 SUMMARY: - the along-track of reference (i.e., for comparison) is **Geosat2**. In this case study #2, this along-track **is NOT included** in the mapping. - the SSH_{map} are generated from **3 nadir altimeters** TPN, J1, EN with the DUACS system - we focus on the region in the north-atlantic bassin **(10řx10ř box centered near 330řE-44řN)** - this case study is similar to the approach used in the manuscript

Note that the ratio PSD(mapping_error)/PSD(SSH_{model}) can gives access to the estimation of the map resolution. However, the DUACS maps and along-track products gives us access to only: PSD(SSH_{map}), PSD(SSH_{obs}) and PSD(intrumental_error). We will have to play with this three latter quantities to estimate the resolution. In the study cases 1 and 2, we shows that the PSD(SSH_{map}), PSD(SSH_{obs}) and PSD(intrumental_error) can estimate the resolution and that it is in good agreement with the ratio PSD(mapping_error)/PSD(SSH_{model}).

PYTHON MODULE

```
In [1]: from netCDF4 import Dataset
        import numpy as np
        import matplotlib.pyplot as plt
        from scipy import signal, interpolate
        from matplotlib.ticker import FormatStrFormatter
```

READ THE SSH SEGMENTS

Note: The SSH segments were generated using same algorithm as used for the study described in manuscript. There are 800km long, referenced by their mean latitude and mean longitude and they overlap over 25%

NOTATION

- **SSH_{model}**: alongtrack SSH from model
- **SSH_{obs}**: alongtrack SSH from model including white noise

- **SSH**$_{map}$: SSH mapped with the DUACS algorithm using 3 altimeters data, and interpolated onto the alongtrack SSH$_{model}$ path
- **intrumental_error**: SSH$_{obs}$ minus SSH$_{model}$
- **mapping_error**: SSH$_{model}$ minus SSH$_{map}$

In [2]:
```
nc = Dataset('./data/dataset_independent_alongtrack_G2.nc','r')
ssh_model = nc.variables['ssh_model'][:]
ssh_obs = nc.variables['ssh_obs'][:]
ssh_map = nc.variables['ssh_map'][:]
dx = nc.variables['resolution'][:]
nperseg = nc.dimensions['segment_size'].size
nc.close()
```

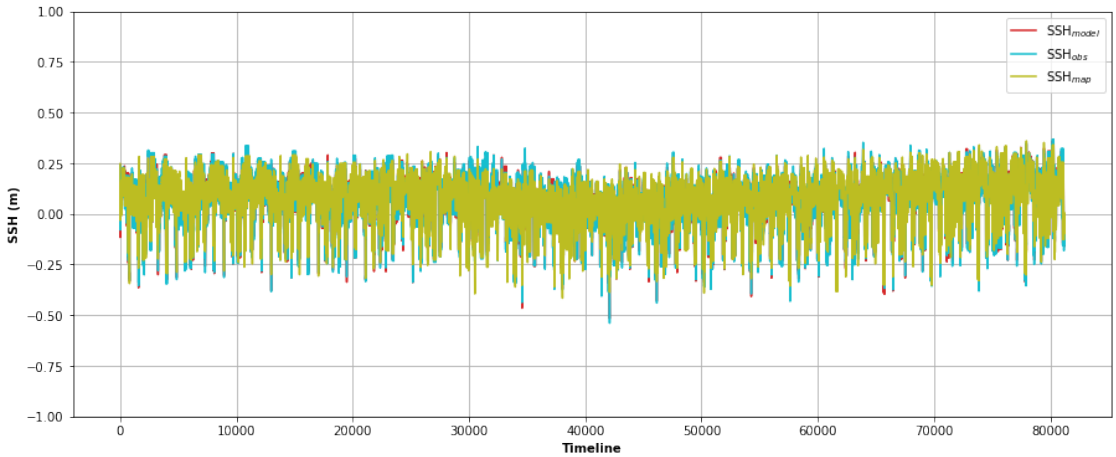
COMPUTE INSTRUMENTAL ERROR & MAPPING ERROR

In [3]:
```
intrumental_error = ssh_obs - ssh_model
mapping_error = ssh_model - ssh_map
```

QUANTITY THAT CAN BE EVALUATED FROM ALTIMETRY DATA : SSH$_{map}$ - SSH$_{obs}$

In [4]:
```
# What can evaluated from altimetry product (we don't have access to ssh_model)
ssh_obs_minus_map = ssh_obs - ssh_map
```

In [5]:
```
plt.figure(figsize=(15, 6))
plt.xlabel("Timeline", fontweight='bold')
plt.ylabel("SSH (m)", fontweight='bold')
plt.plot(ssh_model, c='C3', label='SSH$_{model}$')
plt.plot(ssh_obs, c='C9', label='SSH$_{obs}$')
plt.plot(ssh_map, c='C8', label='SSH$_{map}$')
plt.ylim(-1, 1)
plt.legend(loc='best')
plt.grid()
plt.show()
```

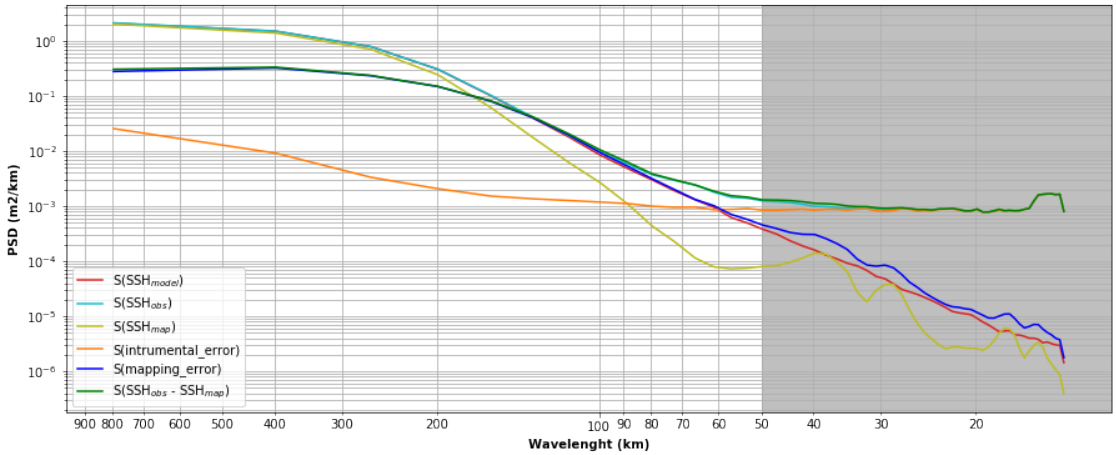


SPECTRAL CONTENT OF EACH SIGNAL

In [6]:
```
freq, psd_ssh_model = signal.welch(ssh_model,
                                   fs=1/dx,
                                   nperseg=nperseg,
                                   scaling='density',
                                   noverlap=0)

_, psd_ssh_obs = signal.welch(ssh_obs,
                              fs=1/dx,
                              nperseg=nperseg,
                              scaling='density',
                              noverlap=0)

_, psd_ssh_map = signal.welch(ssh_map,
                              fs=1/dx,
                              nperseg=nperseg,
                              scaling='density',
                              noverlap=0)

_, psd_intrumental_error = signal.welch(intrumental_error,
                                        fs=1/dx,
                                        nperseg=nperseg,
                                        scaling='density',
                                        noverlap=0)

_, psd_mapping_error = signal.welch(mapping_error,
                                    fs=1/dx,
                                    nperseg=nperseg,
                                    scaling='density',
                                    noverlap=0)

_, psd_ssh_obs_minus_map = signal.welch(ssh_obs_minus_map,
                                        fs=1/dx,
                                        nperseg=nperseg,
                                        scaling='density',
                                        noverlap=0)
```

In [7]:
```
fig, ax = plt.subplots(figsize=(15, 6))
plt.plot(1/freq, psd_ssh_model, c='C3', label='S(SSH$_{model}$)')
plt.plot(1/freq, psd_ssh_obs, c='C9', label='S(SSH$_{obs}$)')
plt.plot(1/freq, psd_ssh_map, c='C8', label='S(SSH$_{map}$)')
plt.plot(1./freq, psd_intrumental_error, c='C1', label='S(intrumental_error)')
plt.plot(1./freq, psd_mapping_error, c='b', label='S(mapping_error)')
plt.plot(1./freq, psd_ssh_obs_minus_map, c='g', label='S(SSH$_{obs}$ - SSH$_{map}$)')
plt.gca().invert_xaxis()
plt.yscale('log')
plt.xscale('log')
plt.xlabel('Wavelenght (km)', fontweight='bold')
plt.ylabel('PSD (m2/km)', fontweight='bold')
```

```
plt.legend(loc='best')
plt.grid(which='both')
ax.axvspan(0, 50, color='grey', alpha=0.5)
ax.xaxis.set_major_formatter(FormatStrFormatter('%.f'))
ax.xaxis.set_minor_formatter(FormatStrFormatter('%.f'))
plt.show()
```



The figure represents the power spectral densities from each type of signal. It illustrates a good agreement of the power spectral density S(SSH$_{obs}$), S(SSH$_{map}$) and S(SSH$_{model}$) for wavelength > 200-300 km. For wavelength < 200 km the power spectral density of the SSH$_{map}$ is less energetic than for the SSH$_{obs}$, linked to the filtering properties of te DUACS mapping system.

The signal-noise-ratio for the input along-track is between 50-60km (crossing of the red and orange lines), setting the along-track resolution limit as defined in Dufau et al (2016).

For wavelenght greater than this along-track resolution limit, the power spectral density of the mapping error S(mapping_error) is larger than the power spectral density of the instrumental error S(instrumental_error).

The power spectral density S(SSH$_{map}$) for wavelenght smaller than this along-track resolution limit (approx. grey area) is below the grid spacing of the DUACS maps. The power spectral density S(SSH$_{map}$) below this limit is hence resulting from the interpolation of the SSH$_{map}$ onto the alongtrack position

CROSS SPECTRUM

```
In [8]: freq, cross_psd_ssh_map_ssh_obs = signal.csd(ssh_map,
                                                    ssh_obs,
                                                    fs=1./dx,
                                                    nperseg=nperseg,
                                                    noverlap=0)
```

SPECTRAL COHERENCE

```
In [9]: # Compute coherence ssh_map ssh_obs
        freq, coh_ssh_map_ssh_obs = signal.coherence(ssh_map,
```

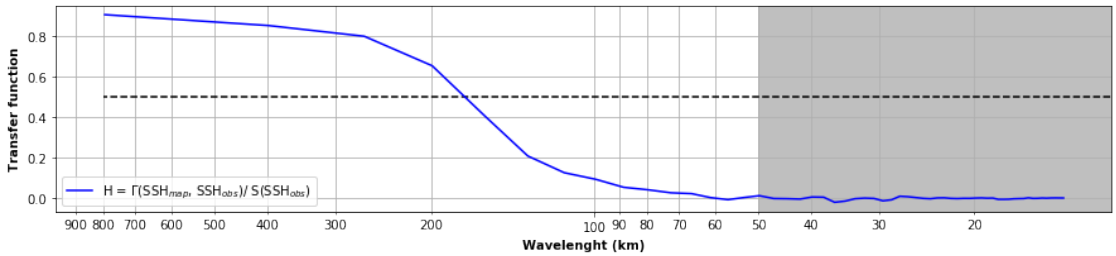
```
                                                    ssh_obs,
                                                    fs=1/dx,
                                                    nperseg=nperseg,
                                                    noverlap=0)

        # Compute coherence ssh_map ssh_model
        _, coh_ssh_map_ssh_model = signal.coherence(ssh_map,
                                                    ssh_model,
                                                    fs=1/dx,
                                                    nperseg=nperseg,
                                                    noverlap=0)
```

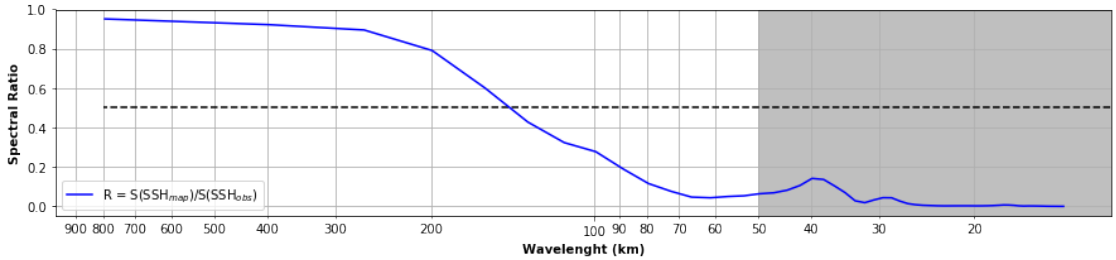
In [10]:
```
        # TRANSFER FUNCTION
        fig, ax = plt.subplots(figsize=(15, 3))
        plt.plot(1./freq, cross_psd_ssh_map_ssh_obs/psd_ssh_obs,
                 c='b', label='H = $\Gamma$(SSH$_{map}$, SSH$_{obs}$)/ S(SSH$_{obs}$)')
        plt.xscale('log')
        plt.gca().invert_xaxis()
        ax.xaxis.set_major_formatter(FormatStrFormatter('%.f'))
        ax.xaxis.set_minor_formatter(FormatStrFormatter('%.f'))
        plt.xlabel('Wavelenght (km)', fontweight='bold')
        plt.ylabel('Transfer function', fontweight='bold')
        plt.hlines(0.5, xmin=0, xmax=800, linestyle='--')
        plt.legend()
        ax.axvspan(0, 50, color='grey', alpha=0.5)
        plt.grid(which='both')

        # SPECTRAL RATIO
        fig, ax = plt.subplots(figsize=(15, 3))
        plt.plot(1./freq, psd_ssh_map/psd_ssh_obs, c='b',
                 label='R = S(SSH$_{map}$)/S(SSH$_{obs}$)')
        plt.xscale('log')
        plt.gca().invert_xaxis()
        ax.xaxis.set_major_formatter(FormatStrFormatter('%.f'))
        ax.xaxis.set_minor_formatter(FormatStrFormatter('%.f'))
        plt.xlabel('Wavelenght (km)', fontweight='bold')
        plt.ylabel('Spectral Ratio', fontweight='bold')
        plt.hlines(0.5, xmin=0, xmax=800, linestyle='--')
        plt.legend()
        ax.axvspan(0, 50, color='grey', alpha=0.5)
        plt.grid(which='both')

        # MAGNITUDE SQUARED COHERENCE
        fig, ax = plt.subplots(figsize=(15, 3))
        plt.plot(1./freq, coh_ssh_map_ssh_obs, c='C8',
                 label='$\gamma^2$(SSH$_{map}$, SSH$_{obs}$)', lw=6)
        plt.plot(1./freq, coh_ssh_map_ssh_model, c='b',
                 label='$\gamma^2$(SSH$_{map}$, SSH$_{model}$)')
```

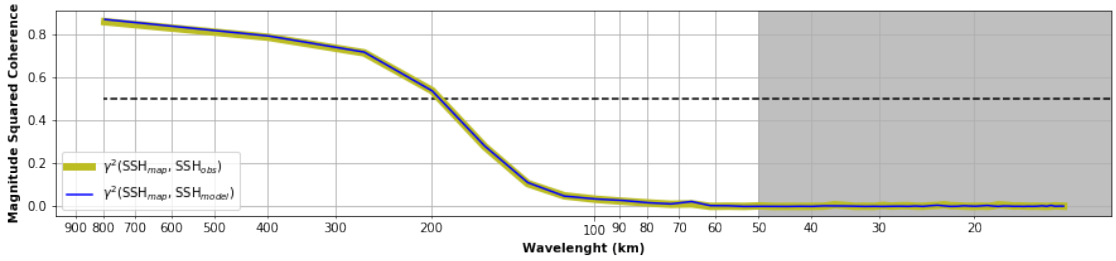
```
plt.hlines(0.5, xmin=0, xmax=800, linestyle='--')
plt.gca().invert_xaxis()
plt.grid(which='both')
plt.legend()
plt.xscale('log')
ax.xaxis.set_major_formatter(FormatStrFormatter('%.f'))
ax.xaxis.set_minor_formatter(FormatStrFormatter('%.f'))
ax.axvspan(0, 50, color='grey', alpha=0.5)
plt.xlabel('Wavelenght (km)', fontweight='bold')
plt.ylabel('Magnitude Squared Coherence', fontweight='bold')

# RATIO ERROR SPECTRUM / (SSH_OBS SPECTRUM)
fig, ax = plt.subplots(figsize=(15, 3))
plt.hlines(0.5, xmin=0, xmax=800, linestyle='--')
plt.plot(1./freq, (psd_mapping_error)/psd_ssh_model, c='r',
        label='S(mapping_error)/S(SSH$_{model}$)')
plt.plot(1./freq, (psd_ssh_obs_minus_map)/(psd_ssh_obs), c='b',
         label='S(SSH$_{obs}$ - SSH$_{map}$)/S((SSH$_{obs}$)')
plt.plot(1./freq, (psd_ssh_obs_minus_map - psd_intrumental_error)/
         (psd_ssh_obs - psd_intrumental_error), c='C8',
         label='(S(SSH$_{obs}$ - SSH$_{map}$) - S(intrumental_error))/(S(SSH$_{obs}$)

percent = 0.5

upper_bound = np.maximum((psd_ssh_obs_minus_map -
                          (1 - percent)*psd_intrumental_error)/
                (psd_ssh_obs - (1 - percent)*psd_intrumental_error),
                         np.abs(psd_ssh_obs_minus_map -
                          (1 + percent)*psd_intrumental_error)/
                (psd_ssh_obs - (1 + percent)*psd_intrumental_error))

lower_bound = np.minimum((psd_ssh_obs_minus_map -
                          (1 - percent)*psd_intrumental_error)/
                (psd_ssh_obs - (1 - percent)*psd_intrumental_error),
                        (psd_ssh_obs_minus_map -
                         np.abs(1 + percent)*psd_intrumental_error)/
                (psd_ssh_obs - (1 + percent)*psd_intrumental_error))

ax.fill_between(1./freq, lower_bound, upper_bound, color='C8', alpha=0.5)
plt.ylim(0,2)
plt.gca().invert_xaxis()
plt.grid(which='both')
plt.legend()
plt.xscale('log')
ax.axvspan(0, 50, color='grey', alpha=0.5)
ax.xaxis.set_major_formatter(FormatStrFormatter('%.f'))
ax.xaxis.set_minor_formatter(FormatStrFormatter('%.f'))
plt.xlabel('Wavelenght (km)', fontweight='bold')
```

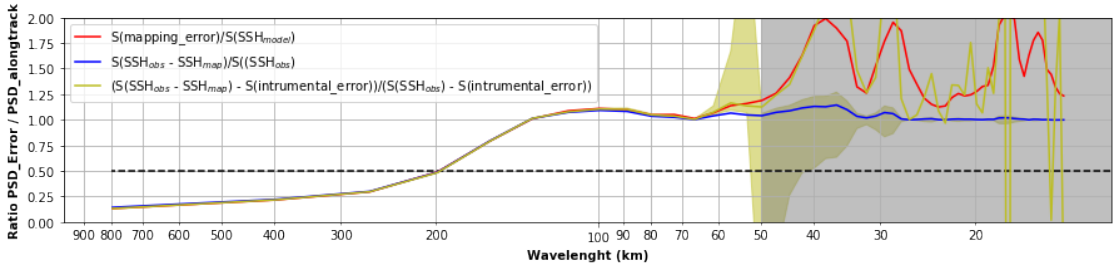
```
        plt.ylabel('Ratio PSD_Error / PSD_alongtrack', fontweight='bold')

        plt.show()
```

```
/anaconda3/lib/python3.7/site-packages/numpy/core/numeric.py:501: ComplexWarning: Casting compl
  return array(a, dtype, copy=False, order=order)
```









CONCLUSIONS

We here used "realistic" dataset to investigate the response of various spectral metrics as a solution for estimating the resolution of the DUACS gridded products. The results for this case study #1 sligthly differ from case study #2 where the reference along-track dataset is used in the mapping process. The transfer function, the magnitude squared coherence and the ratio PSD error, PSD signal are affected by the independency/NON-indepedendy of the reference dataset. The spectal ratio PSD map / PSD alongtrack is weakly affected. Obviously, when the reference data is non-independent it "forces" the SSH map to be closer to the SSH obs, hence the curves of the tranfer function, the magnitude squared coherence and the ratio PSD Error / PS_alongtrack are shifted toward higher wavelenght.

Our main conclusions are:

1) The estimation of the spectral metrics are affected by considering or not the reference data as independent. Indepedent data lead to coarser resolution (spectral metrics are shifted towards higher wavenumber)

2) Magnitude squared coherence and Ratio PSD error/PSD along-track are in good agreements, meaning the phase consistency is the dominant factor controlling the "quality" of the DUACS maps

3) Spectral ratio is weakly affected by considering or not the reference dataset as independent

4) Considering noise or not in the alongtrack dataset as weak impact on the coherence (probably linked to the fact that mapping error >> intrumental error at wavelength that we are looking at.

5) The theoretical basis for the coherence-based measure of resolution at MSC=0.5 was probably mis-interpreted. It might be clearer to defined it as the coherent/incoherent threshold.

6) However, based on Tom's comments and also due to the fact that the coherence based measure is insensitive to the difference in the signals amplitudes, an other approach based on the spectral ratio of the error over signal is preferable for estimating the resolution of the DUACS maps. By defining the resolution as the 0.5 threshold of the ratio PSD(Error)/PSD(ref) (limit where there is two times more signal than error) we shows that for both case study, the estimation of this ratio as S(mapping_error)/S(SSH_{model}) or S((SSH_{map} - (SSH_{obs})/S((SSH_{obs}) are similar for wavelenght > 50 km.

Tom's review elegantly shows the behaviour of the spectral metrics in various scenarios of noise level (no noise, high noise , low noise level). It is hence an assessment of the resolution in a general context. Our study with realistic data is more specific (applied in altimetry data processing context).

Example_OSSE_obs_IN_mapping_NATL60

April 18, 2019

0.1 CASE STUDY 2: COMPARISON WITH NON-INDEPENDENT ALONG-TRACK

INTRODUCTION

We here propose to show the behaviour of the spectral analysis in an OSSE case study based on NATL60 simulation. The input comes from hourly SSH data from a 1 year long period of the run NATL60-CJM165 made at IGE (Grenoble). We used the SWOTsimulator to sample alongtrack data (nadir-like dataset) on this simulation. The constellation is similar to the 20030101-20031231 period: Jason 1, Envisat, Geosat2, Topex/Poseidon interleaved. SWOTsimulator gives us the true SSH (SSH_{model}) and a pseudo-obs SSH (SSH_{obs} = SSH_{model} + white noise). These pseudo-obs SSH_{obs} are then used in the DUACS mapping.

CASE STUDY #2 SUMMARY: - the along-track of reference (i.e., for comparison) is **Jason1**. In this case study #2, this along-track **is included** in the mapping. - the SSH_{map} produced are generated from **3 nadir altimeters** TPN, J1, EN with the DUACS system - we focus on the region in the north-atlantic bassin **(10řx10ř box centered near 330řE-44řN)**

Note that the ratio PSD(mapping_error)/PSD(SSH_{model}) can gives access to the estimation of the map resolution. However, the DUACS maps and along-track products gives us access to only: PSD(SSH_{map}), PSD(SSH_{obs}) and PSD(intrumental_error). We will have to play with this three latter quantities to estimate the resolution. In the study cases 1 and 2, we shows that the PSD(SSH_{map}), PSD(SSH_{obs}) and PSD(intrumental_error) can estimate the resolution and that it is in good agreement with the ratio PSD(mapping_error)/PSD(SSH_{model}).

PYTHON MODULE

```
In [1]: from netCDF4 import Dataset
        import numpy as np
        import matplotlib.pyplot as plt
        from scipy import signal, interpolate
        from matplotlib.ticker import FormatStrFormatter
```

READ THE SSH SEGMENTS

Note: The SSH segments were generated using same algorithm as used for the study described in the manuscript. There are 800km long, referenced by their mean latitude and mean longitude and they overlap over 25%

NOTATION

- **SSH_{model}**: alongtrack SSH from model
- **SSH_{obs}**: alongtrack SSH from model including white noise
- **SSH_{map}**: SSH mapped with the DUACS algorithm using 3 altimeters data, and interpolated onto the alongtrack SSH_{model} path

- **intrumental_error**: SSH_{obs} minus SSH_{model}
- **map_error**: SSH_{model} minus SSH_{map}

```
In [2]: nc = Dataset('./data/dataset_NON_independent_alongtrack_J1.nc','r')
        ssh_model = nc.variables['ssh_model'][:]
        ssh_obs = nc.variables['ssh_obs'][:]
        ssh_map = nc.variables['ssh_map'][:]
        dx = nc.variables['resolution'][:]
        nperseg = nc.dimensions['segment_size'].size
        nc.close()
```

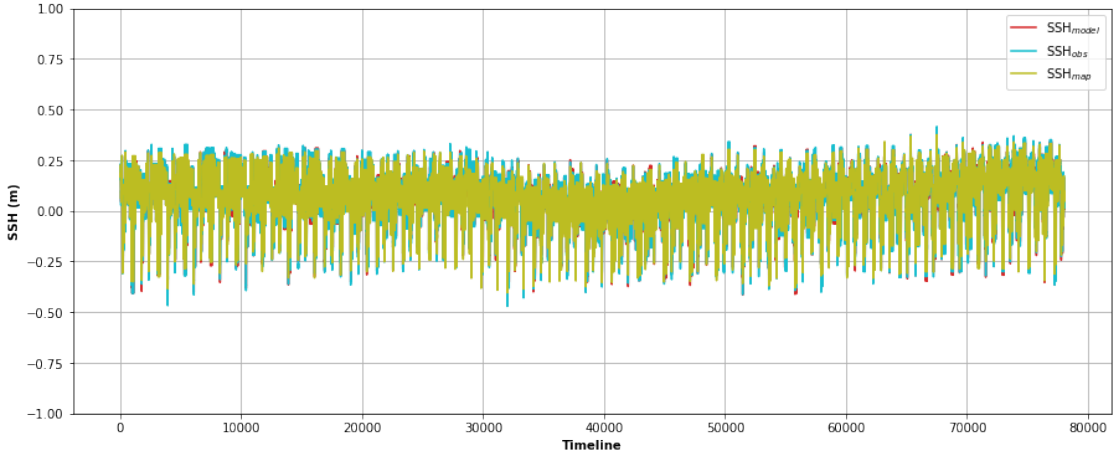
COMPUTE INSTRUMENTAL ERROR & MAPPING ERROR

```
In [3]: intrumental_error = ssh_obs - ssh_model
        mapping_error = ssh_model - ssh_map
```

QUANTITY THAT CAN BE EVALUATED FROM ALTIMETRY DATA : SSH_{map} - SSH_{obs}

```
In [4]: # What can evaluated from altimetry product (we don't have access to ssh_model)
        psd_ssh_obs_minus_map = ssh_obs - ssh_map
```

```
In [5]: plt.figure(figsize=(15, 6))
        plt.xlabel("Timeline", fontweight='bold')
        plt.ylabel("SSH (m)", fontweight='bold')
        plt.plot(ssh_model, c='C3', label='SSH$_{model}$')
        plt.plot(ssh_obs, c='C9', label='SSH$_{obs}$')
        plt.plot(ssh_map, c='C8', label='SSH$_{map}$')
        plt.ylim(-1, 1)
        plt.legend(loc='best')
        plt.grid()
        plt.show()
```

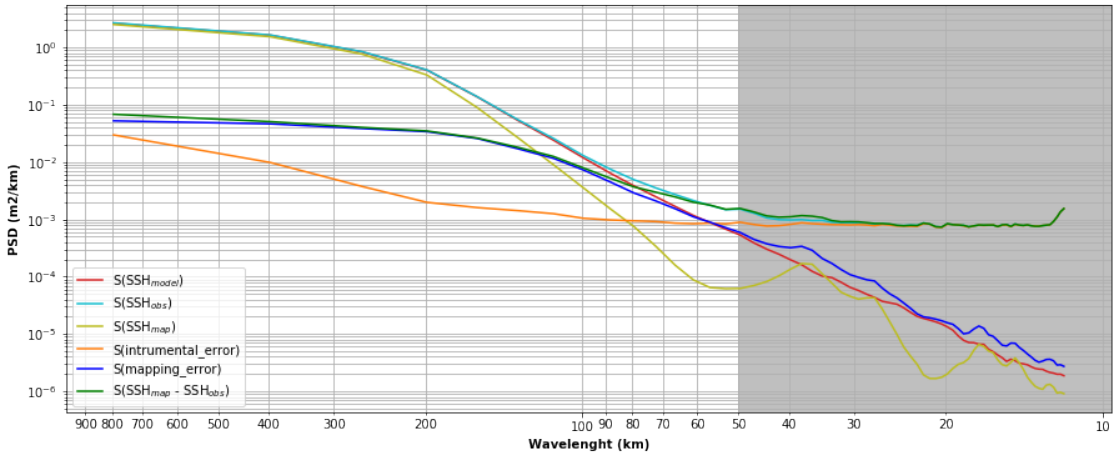


SPECTRAL CONTENT OF EACH SIGNAL

In [6]:
```
freq, psd_ssh_model = signal.welch(ssh_model,
                                   fs=1/dx,
                                   nperseg=nperseg,
                                   scaling='density',
                                   noverlap=0)

_, psd_ssh_obs = signal.welch(ssh_obs,
                              fs=1/dx,
                              nperseg=nperseg,
                              scaling='density',
                              noverlap=0)

_, psd_ssh_map = signal.welch(ssh_map,
                              fs=1/dx,
                              nperseg=nperseg,
                              scaling='density',
                              noverlap=0)

_, psd_intrumental_error = signal.welch(intrumental_error,
                                        fs=1/dx,
                                        nperseg=nperseg,
                                        scaling='density',
                                        noverlap=0)

_, psd_mapping_error = signal.welch(mapping_error,
                                    fs=1/dx,
                                    nperseg=nperseg,
                                    scaling='density',
                                    noverlap=0)

_, psd_ssh_obs_minus_map = signal.welch(psd_ssh_obs_minus_map,
                                        fs=1/dx,
                                        nperseg=nperseg,
                                        scaling='density',
                                        noverlap=0)
```

In [7]:
```
fig, ax = plt.subplots(figsize=(15, 6))
plt.plot(1/freq, psd_ssh_model, c='C3', label='S(SSH$_{model}$)')
plt.plot(1/freq, psd_ssh_obs, c='C9', label='S(SSH$_{obs}$)')
plt.plot(1/freq, psd_ssh_map, c='C8', label='S(SSH$_{map}$)')
plt.plot(1./freq, psd_intrumental_error, c='C1', label='S(intrumental_error)')
plt.plot(1./freq, psd_mapping_error, c='b', label='S(mapping_error)')
plt.plot(1./freq, psd_ssh_obs_minus_map, c='g', label='S(SSH$_{map}$ - SSH$_{obs}$)')
ax.axvspan(0, 50, color='grey', alpha=0.5)
plt.gca().invert_xaxis()
plt.yscale('log')
plt.xscale('log')
plt.xlabel('Wavelenght (km)', fontweight='bold')
```

```
plt.ylabel('PSD (m2/km)', fontweight='bold')
plt.legend(loc='best')
plt.grid(which='both')
ax.xaxis.set_major_formatter(FormatStrFormatter('%.f'))
ax.xaxis.set_minor_formatter(FormatStrFormatter('%.f'))
plt.show()
```



The figure represents the power spectral densities from each type of signal. It illustrates a good agreement of the power spectral density S(SSH$_{obs}$), S(SSH$_{map}$) and S(SSH$_{model}$) for wavelength > 200-300 km. For wavelength < 200 km the power spectral density of the SSH$_{map}$ is less energetic than for the SSH$_{obs}$, linked to the filtering properties of te DUACS mapping system.

The signal-noise-ratio for the input along-track is between 50-60km (crossing of the red and orange lines), setting the along-track resolution limit as defined in Dufau et al (2016).

For wavelenght greater than this along-track resolution limit, the power spectral density of the mapping error S(mapping_error) is larger than the power spectral density of the instrumental error S(instrumental_error).

The power spectral density S(SSH$_{map}$) for wavelenght smaller than this along-track resolution limit (approx. grey area) is below the grid spacing of the DUACS maps. The power spectral density S(SSH$_{map}$) below this limit is hence resulting from the interpolation of the SSH$_{map}$ onto the reference alongtrack position

CROSS SPECTRUM

```
In [8]: freq, cross_psd_ssh_map_ssh_obs = signal.csd(ssh_map,
                                                    ssh_obs,
                                                    fs=1./dx,
                                                    nperseg=nperseg,
                                                    noverlap=0)
```

SPECTRAL COHERENCE

```
In [9]: # Compute coherence ssh_map ssh_obs
        freq, coh_ssh_map_ssh_obs = signal.coherence(ssh_map,
```

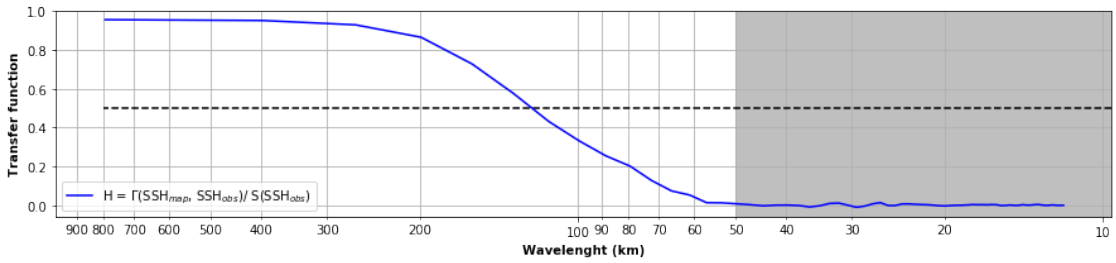
```
                                                    ssh_obs,
                                                    fs=1/dx,
                                                    nperseg=nperseg,
                                                    noverlap=0)

        # Compute coherence ssh_map ssh_model
        _, coh_ssh_map_ssh_model = signal.coherence(ssh_map,
                                                    ssh_model,
                                                    fs=1/dx,
                                                    nperseg=nperseg,
                                                    noverlap=0)
```

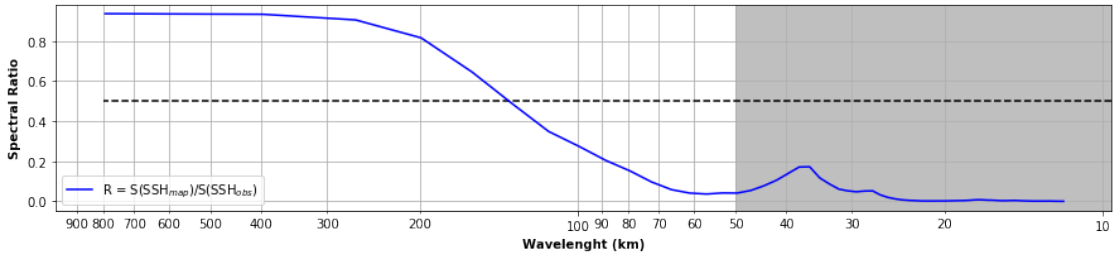
In [10]:
```
        # TRANSFER FUNCTION
        fig, ax = plt.subplots(figsize=(15, 3))
        plt.plot(1./freq, cross_psd_ssh_map_ssh_obs/psd_ssh_obs,
                 c='b', label='H = $\Gamma$(SSH$_{map}$, SSH$_{obs}$)/ S(SSH$_{obs}$)')
        plt.xscale('log')
        plt.gca().invert_xaxis()
        ax.xaxis.set_major_formatter(FormatStrFormatter('%.f'))
        ax.xaxis.set_minor_formatter(FormatStrFormatter('%.f'))
        plt.xlabel('Wavelenght (km)', fontweight='bold')
        plt.ylabel('Transfer function', fontweight='bold')
        plt.hlines(0.5, xmin=0, xmax=800, linestyle='--')
        plt.legend()
        ax.axvspan(0, 50, color='grey', alpha=0.5)
        plt.grid(which='both')

        # SPECTRAL RATIO
        fig, ax = plt.subplots(figsize=(15, 3))
        plt.plot(1./freq, psd_ssh_map/psd_ssh_obs, c='b',
                 label='R = S(SSH$_{map}$)/S(SSH$_{obs}$)')
        plt.xscale('log')
        plt.gca().invert_xaxis()
        ax.xaxis.set_major_formatter(FormatStrFormatter('%.f'))
        ax.xaxis.set_minor_formatter(FormatStrFormatter('%.f'))
        plt.xlabel('Wavelenght (km)', fontweight='bold')
        plt.ylabel('Spectral Ratio', fontweight='bold')
        plt.hlines(0.5, xmin=0, xmax=800, linestyle='--')
        plt.legend()
        ax.axvspan(0, 50, color='grey', alpha=0.5)
        plt.grid(which='both')

        # MAGNITUDE SQUARED COHERENCE
        fig, ax = plt.subplots(figsize=(15, 3))
        plt.plot(1./freq, coh_ssh_map_ssh_obs, c='C8',
                 label='$\gamma^2$(SSH$_{map}$, SSH$_{obs}$)', lw=6)
        plt.plot(1./freq, coh_ssh_map_ssh_model, c='b',
                 label='$\gamma^2$(SSH$_{map}$, SSH$_{model}$)')
```

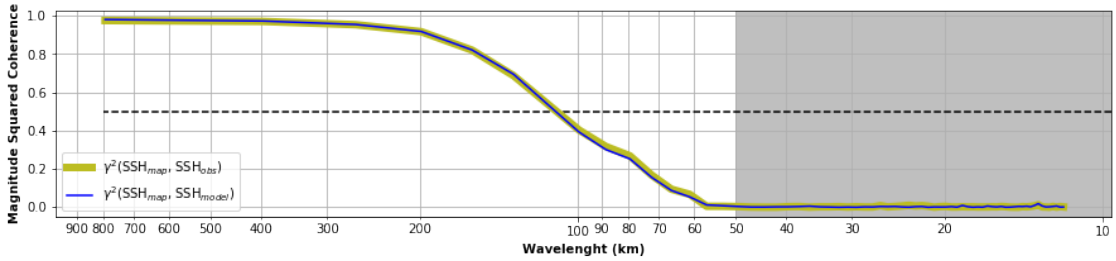
```
plt.hlines(0.5, xmin=0, xmax=800, linestyle='--')
plt.gca().invert_xaxis()
plt.grid(which='both')
plt.legend()
plt.xscale('log')
ax.xaxis.set_major_formatter(FormatStrFormatter('%.f'))
ax.xaxis.set_minor_formatter(FormatStrFormatter('%.f'))
ax.axvspan(0, 50, color='grey', alpha=0.5)
plt.xlabel('Wavelenght (km)', fontweight='bold')
plt.ylabel('Magnitude Squared Coherence', fontweight='bold')

# RATIO ERROR SPECTRUM / (SSH_OBS SPECTRUM)
fig, ax = plt.subplots(figsize=(15, 3))
plt.hlines(0.5, xmin=0, xmax=800, linestyle='--')
plt.plot(1./freq, (psd_mapping_error)/psd_ssh_model, c='r',
        label=' S(mapping_error)/S(SSH$_{model}$)')
plt.plot(1./freq, (psd_ssh_obs_minus_map)/(psd_ssh_obs), c='b',
         label='S((SSH$_{map}$ - (SSH$_{obs}$)/S((SSH$_{obs}$)')
#plt.plot(1./freq, (psd_ssh_obs_minus_map - psd_intrumental_error)/
#         (psd_ssh_obs - psd_intrumental_error), c='C8',
#         label='(S(SSH$_{map}$ - SSH$_{obs}$) - S(intrumental_error))/(S(SSH$_{obs}$

percent = 0.5

#percent = 0.5
upper_bound = np.maximum((psd_ssh_obs_minus_map -
                          (1 - percent)*psd_intrumental_error)/
                (psd_ssh_obs - (1 - percent)*psd_intrumental_error),
                         (psd_ssh_obs_minus_map -
                          (1 + percent)*psd_intrumental_error)/
                (psd_ssh_obs - (1 + percent)*psd_intrumental_error))
lower_bound = np.minimum((psd_ssh_obs_minus_map -
                          (1 - percent)*psd_intrumental_error)/
                (psd_ssh_obs - (1 - percent)*psd_intrumental_error),
                         (psd_ssh_obs_minus_map -
                          (1 + percent)*psd_intrumental_error)/
                (psd_ssh_obs - (1 + percent)*psd_intrumental_error))

ax.fill_between(1./freq, lower_bound, upper_bound, color='C8', alpha=0.5)
plt.ylim(0,2)
plt.gca().invert_xaxis()
plt.grid(which='both')
plt.legend()
plt.xscale('log')
ax.axvspan(0, 50, color='grey', alpha=0.5)
ax.xaxis.set_major_formatter(FormatStrFormatter('%.f'))
ax.xaxis.set_minor_formatter(FormatStrFormatter('%.f'))
plt.xlabel('Wavelenght (km)', fontweight='bold')
```

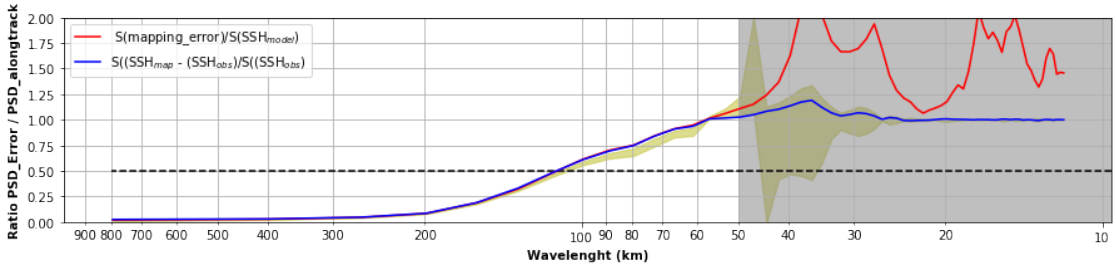
```
        plt.ylabel('Ratio PSD_Error / PSD_alongtrack', fontweight='bold')
        plt.show()
```

```
/anaconda3/lib/python3.7/site-packages/numpy/core/numeric.py:501: ComplexWarning: Casting compl
  return array(a, dtype, copy=False, order=order)
```









This figure represents various metrics to estimate the resolution the map products: the transfer function, the spectral ratio between SSH_{map} and SSH_{obs}, the Magnitude Squared Coherence and the ratio PSD error / PSD SSH_{obs}. The threshold γ^2 = 0.5 and Q = 0.5 gives relatively similar resolution limit for the maps.

The yellow envelop in lower panel gives the spread of the ratio (S(SSH_{map} - SSH_{obs}) - α.S(intrumental_error))/(S(SSH_{obs}) - α.S(intrumental_error)) for $\alpha = \pm$ 50% increased/decreased instrumental error level. It shows the sensitivity of the ratio to the noise level in the reference along-track dataset. For wavelenght > 50 km, the ratio is weakly impacted by the amount of nosie level, and the three curve are rather similar.

Example_OSSE_obs_NOT_mapping_NATL60_high_instrumental_nois

April 18, 2019

0.1 CASE STUDY 3: INDEPENDENT ALONG-TRACK WITH HIGHER INSTRUMENTAL NOISE

INTRODUCTION

We here propose to show the behaviour of the spectral analysis in an OSSE case study based on NATL60 simulation. The input comes from hourly SSH data from a 1 year long period of the run NATL60-CJM165 made at IGE (Grenoble). We used the SWOTsimulator to sample alongtrack data (nadir-like dataset) on this simulation. The constellation is similar to the 20030101-20031231 period: Jason 1, Envisat, Geosat2, Topex/Poseidon interleaved. SWOTsimulator gives us the true SSH (SSH_{model}) and a pseudo-obs SSH (SSH_{obs} = SSH_{model} + white noise). These pseudo-obs SSH_{obs} are then used in the DUACS mapping.

CASE STUDY #3 SUMMARY: - the along-track of reference (i.e., for comparison) is **Geosat2**. In this case study #3, this along-track **is NOT included** in the mapping. **We virtually add extra noise to this reference along-track to simulate a case where the along-track resolution is above the filtering properties of the DUACS system** - the SSH_{map} are generated from **3 nadir altimeters** TPN, J1, EN with the DUACS system - we focus on the region in the north-atlantic bassin (**10řx10ř box centered near 330řE-44řN**)

PYTHON MODULE

```
In [1]: from netCDF4 import Dataset
        import numpy as np
        import matplotlib.pyplot as plt
        from scipy import signal, interpolate
        from matplotlib.ticker import FormatStrFormatter
```

READ THE SSH SEGMENTS

Note: The SSH segments were generated using same algorithm as for the study described in the manuscript. There are 800km long, referenced by their mean latitude and mean longitude and they overlap over 25%

NOTATION

- **SSH_{model}**: alongtrack SSH from model
- **SSH_{obs}**: alongtrack SSH from model including white noise
- **SSH_{map}**: SSH mapped with the DUACS algorithm using 3 altimeters data, and interpolated onto the alongtrack SSH_{model} path
- **intrumental_error**: SSH_{obs} minus SSH_{model}

- **mapping_error**: SSH_{model} minus SSH_{map}

```
In [2]: nc = Dataset('./data/dataset_independent_alongtrack_G2.nc','r')
        ssh_model = nc.variables['ssh_model'][:]
        ssh_obs = nc.variables['ssh_obs'][:]
        ssh_map = nc.variables['ssh_map'][:]
        dx = nc.variables['resolution'][:]
        nperseg = nc.dimensions['segment_size'].size
        nc.close()
```

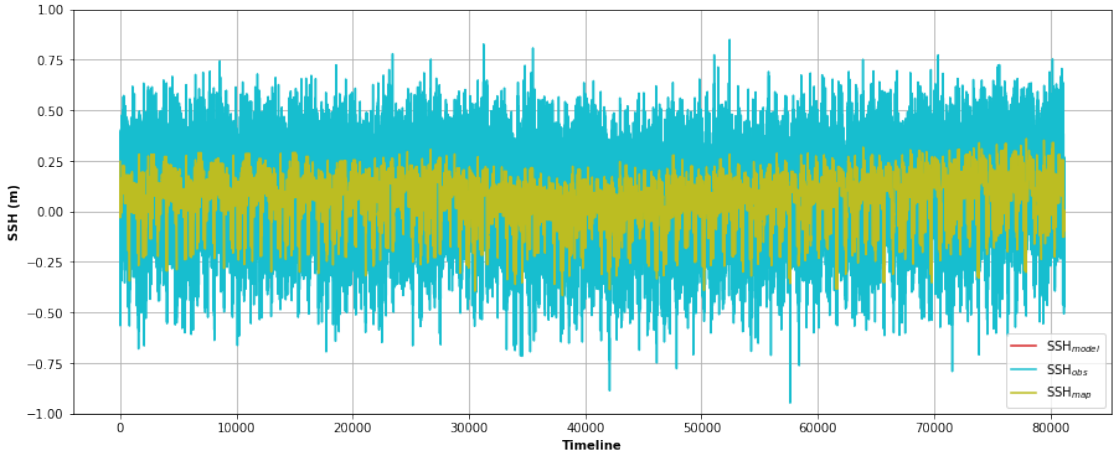
COMPUTE INSTRUMENTAL ERROR & MAPPING ERROR

```
In [3]: # Extra noise
        extra_noise = np.random.normal(0,0.16, ssh_model.size)
        intrumental_error = ssh_obs - ssh_model + extra_noise
        mapping_error = ssh_model - ssh_map
        ssh_obs = intrumental_error + ssh_model
```

QUANTITY THAT CAN BE EVALUATED FROM ALTIMETRY DATA : SSH_{map} - SSH_{obs}

```
In [4]: # What can evaluated from altimetry product (we don't have access to ssh_model)
        ssh_obs_minus_map = ssh_obs - ssh_map
```

```
In [5]: plt.figure(figsize=(15, 6))
        plt.xlabel("Timeline", fontweight='bold')
        plt.ylabel("SSH (m)", fontweight='bold')
        plt.plot(ssh_model, c='C3', label='SSH$_{model}$')
        plt.plot(ssh_obs, c='C9', label='SSH$_{obs}$')
        plt.plot(ssh_map, c='C8', label='SSH$_{map}$')
        plt.ylim(-1, 1)
        plt.legend(loc='best')
        plt.grid()
```

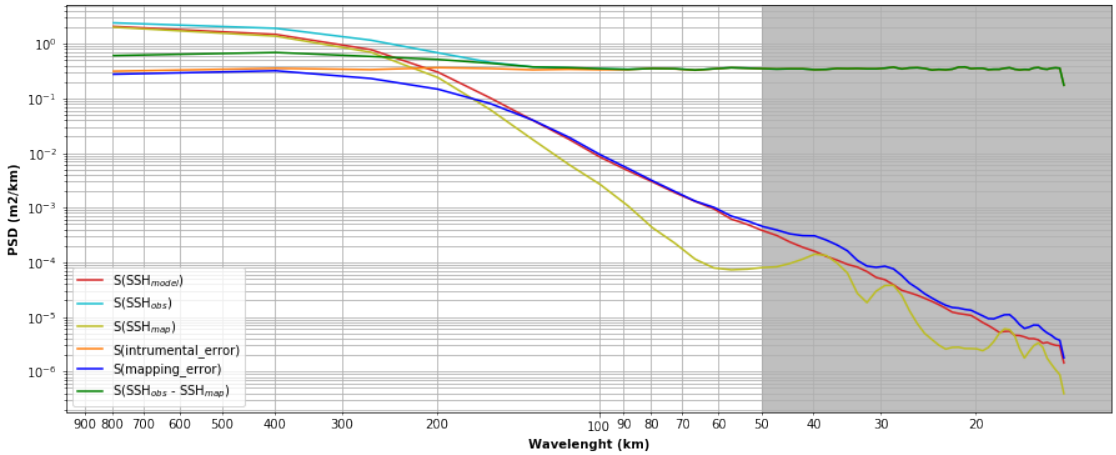


SPECTRAL CONTENT OF EACH SIGNAL

In [6]:
```
freq, psd_ssh_model = signal.welch(ssh_model,
                                   fs=1/dx,
                                   nperseg=nperseg,
                                   scaling='density',
                                   noverlap=0)

_, psd_ssh_obs = signal.welch(ssh_obs,
                              fs=1/dx,
                              nperseg=nperseg,
                              scaling='density',
                              noverlap=0)

_, psd_ssh_map = signal.welch(ssh_map,
                              fs=1/dx,
                              nperseg=nperseg,
                              scaling='density',
                              noverlap=0)

_, psd_intrumental_error = signal.welch(intrumental_error,
                                        fs=1/dx,
                                        nperseg=nperseg,
                                        scaling='density',
                                        noverlap=0)

_, psd_mapping_error = signal.welch(mapping_error,
                                    fs=1/dx,
                                    nperseg=nperseg,
                                    scaling='density',
                                    noverlap=0)

_, psd_ssh_obs_minus_map = signal.welch(ssh_obs_minus_map,
                                        fs=1/dx,
                                        nperseg=nperseg,
                                        scaling='density',
                                        noverlap=0)
```

In [7]:
```
fig, ax = plt.subplots(figsize=(15, 6))
plt.plot(1/freq, psd_ssh_model, c='C3', label='S(SSH$_{model}$)')
plt.plot(1/freq, psd_ssh_obs, c='C9', label='S(SSH$_{obs}$)')
plt.plot(1/freq, psd_ssh_map, c='C8', label='S(SSH$_{map}$)')
plt.plot(1./freq, psd_intrumental_error, c='C1', label='S(intrumental_error)')
plt.plot(1./freq, psd_mapping_error, c='b', label='S(mapping_error)')
plt.plot(1./freq, psd_ssh_obs_minus_map, c='g', label='S(SSH$_{obs}$ - SSH$_{map}$)')
plt.gca().invert_xaxis()
plt.yscale('log')
plt.xscale('log')
plt.xlabel('Wavelenght (km)', fontweight='bold')
plt.ylabel('PSD (m2/km)', fontweight='bold')
```

```
plt.legend(loc='best')
plt.grid(which='both')
ax.axvspan(0, 50, color='grey', alpha=0.5)
ax.xaxis.set_major_formatter(FormatStrFormatter('%.f'))
ax.xaxis.set_minor_formatter(FormatStrFormatter('%.f'))
plt.show()
```



The figure represents the power spectral densities from each type of signal. The agreement of the power spectral density S(SSH$_{obs}$), S(SSH$_{map}$) and S(SSH$_{model}$) is less obvious than in the previous case study.

The signal-noise-ratio for the input along-track is between 200-300km (crossing of the red and orange lines), setting the along-track resolution limit as defined in Dufau et al (2016).

For all wavelenght the power spectral density of the instrumental error S(instrumental_error) is larger than the power spectral density of the mapping error S(mapping_error) .

The wavelenght in the grey area are below the grid spacing of the DUACS maps. The power spectral density S(SSH$_{map}$) below this limit is hence resulting from the interpolation of the SSH$_{map}$ onto the alongtrack position

CROSS SPECTRUM

```
In [8]: freq, cross_psd_ssh_map_ssh_obs = signal.csd(ssh_map,
                                                    ssh_obs,
                                                    fs=1./dx,
                                                    nperseg=nperseg,
                                                    noverlap=0)
```

SPECTRAL COHERENCE

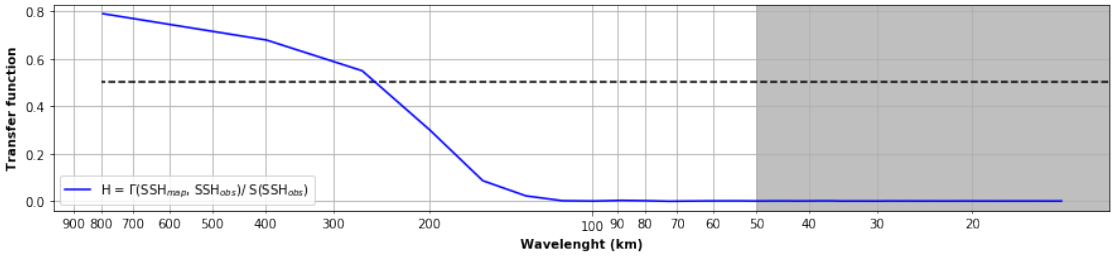
```
In [9]: # Compute coherence ssh_map ssh_obs
        freq, coh_ssh_map_ssh_obs = signal.coherence(ssh_map,
                                                     ssh_obs,
                                                     fs=1/dx,
                                                     nperseg=nperseg,
```

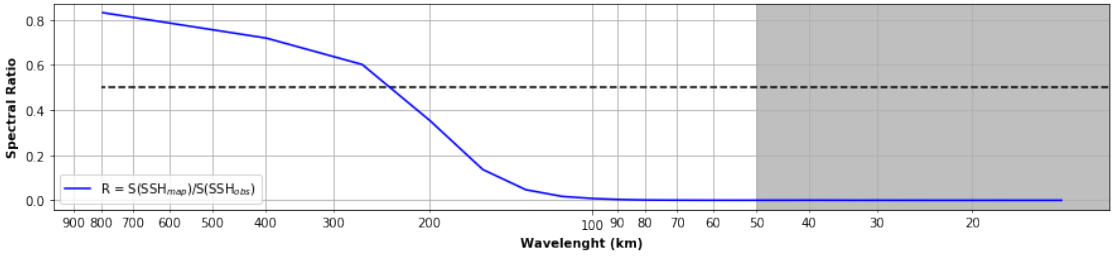
```
                                                    noverlap=0)

        # Compute coherence ssh_map ssh_model
        _, coh_ssh_map_ssh_model = signal.coherence(ssh_map,
                                                    ssh_model,
                                                    fs=1/dx,
                                                    nperseg=nperseg,
                                                    noverlap=0)
```

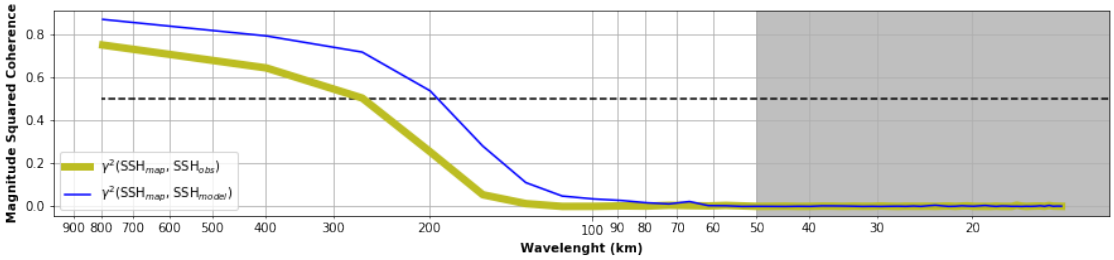
In [10]:
```
        # TRANSFER FUNCTION
        fig, ax = plt.subplots(figsize=(15, 3))
        plt.plot(1./freq, cross_psd_ssh_map_ssh_obs/psd_ssh_obs,
                 c='b', label='H = $\Gamma$(SSH$_{map}$, SSH$_{obs}$)/ S(SSH$_{obs}$)')
        plt.xscale('log')
        plt.gca().invert_xaxis()
        ax.xaxis.set_major_formatter(FormatStrFormatter('%.f'))
        ax.xaxis.set_minor_formatter(FormatStrFormatter('%.f'))
        plt.xlabel('Wavelenght (km)', fontweight='bold')
        plt.ylabel('Transfer function', fontweight='bold')
        plt.hlines(0.5, xmin=0, xmax=800, linestyle='--')
        plt.legend()
        ax.axvspan(0, 50, color='grey', alpha=0.5)
        plt.grid(which='both')

        # SPECTRAL RATIO
        fig, ax = plt.subplots(figsize=(15, 3))
        plt.plot(1./freq, psd_ssh_map/psd_ssh_obs, c='b',
                 label='R = S(SSH$_{map}$)/S(SSH$_{obs}$)')
        plt.xscale('log')
        plt.gca().invert_xaxis()
        ax.xaxis.set_major_formatter(FormatStrFormatter('%.f'))
        ax.xaxis.set_minor_formatter(FormatStrFormatter('%.f'))
        plt.xlabel('Wavelenght (km)', fontweight='bold')
        plt.ylabel('Spectral Ratio', fontweight='bold')
        plt.hlines(0.5, xmin=0, xmax=800, linestyle='--')
        plt.legend()
        ax.axvspan(0, 50, color='grey', alpha=0.5)
        plt.grid(which='both')

        # MAGNITUDE SQUARED COHERENCE
        fig, ax = plt.subplots(figsize=(15, 3))
        plt.plot(1./freq, coh_ssh_map_ssh_obs, c='C8',
                 label='$\gamma^2$(SSH$_{map}$, SSH$_{obs}$)', lw=6)
        plt.plot(1./freq, coh_ssh_map_ssh_model, c='b',
                 label='$\gamma^2$(SSH$_{map}$, SSH$_{model}$)')
        plt.hlines(0.5, xmin=0, xmax=800, linestyle='--')
        plt.gca().invert_xaxis()
        plt.grid(which='both')
```

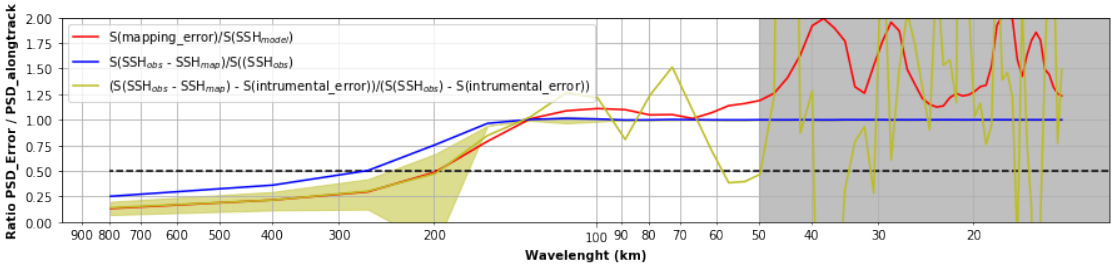
```
plt.legend()
plt.xscale('log')
ax.xaxis.set_major_formatter(FormatStrFormatter('%.f'))
ax.xaxis.set_minor_formatter(FormatStrFormatter('%.f'))
ax.axvspan(0, 50, color='grey', alpha=0.5)
plt.xlabel('Wavelenght (km)', fontweight='bold')
plt.ylabel('Magnitude Squared Coherence', fontweight='bold')

# RATIO ERROR SPECTRUM / (SSH_OBS SPECTRUM)
fig, ax = plt.subplots(figsize=(15, 3))
plt.hlines(0.5, xmin=0, xmax=800, linestyle='--')
plt.plot(1./freq, (psd_mapping_error)/psd_ssh_model, c='r',
        label='S(mapping_error)/S(SSH$_{model}$)')
plt.plot(1./freq, (psd_ssh_obs_minus_map)/(psd_ssh_obs), c='b',
         label='S(SSH$_{obs}$ - SSH$_{map}$)/S((SSH$_{obs}$)')
plt.plot(1./freq, (psd_ssh_obs_minus_map - psd_intrumental_error)/
         (psd_ssh_obs - psd_intrumental_error), c='C8',
         label='(S(SSH$_{obs}$ - SSH$_{map}$) - S(intrumental_error))/(S(SSH$_{obs}$)

percent = 0.5

upper_bound = np.maximum((psd_ssh_obs_minus_map -
                          (1 - percent)*psd_intrumental_error)/
                (psd_ssh_obs - (1 - percent)*psd_intrumental_error),
                         np.abs(psd_ssh_obs_minus_map -
                          (1 + percent)*psd_intrumental_error)/
                (psd_ssh_obs - (1 + percent)*psd_intrumental_error))

lower_bound = np.minimum((psd_ssh_obs_minus_map -
                          (1 - percent)*psd_intrumental_error)/
                (psd_ssh_obs - (1 - percent)*psd_intrumental_error),
                        (psd_ssh_obs_minus_map -
                         np.abs(1 + percent)*psd_intrumental_error)/
                (psd_ssh_obs - (1 + percent)*psd_intrumental_error))

ax.fill_between(1./freq, lower_bound, upper_bound, color='C8', alpha=0.5)
plt.ylim(0,2)
plt.gca().invert_xaxis()
plt.grid(which='both')
plt.legend()
plt.xscale('log')
ax.axvspan(0, 50, color='grey', alpha=0.5)
ax.xaxis.set_major_formatter(FormatStrFormatter('%.f'))
ax.xaxis.set_minor_formatter(FormatStrFormatter('%.f'))
plt.xlabel('Wavelenght (km)', fontweight='bold')
plt.ylabel('Ratio PSD_Error / PSD_alongtrack', fontweight='bold')

plt.show()
```

```
/anaconda3/lib/python3.7/site-packages/numpy/core/numeric.py:501: ComplexWarning: Casting compl
  return array(a, dtype, copy=False, order=order)
```









CONCLUSIONS

Here the spectral ratio corrected from S(intrumental error isrelatively similar to PSD(mapping_error)/PSD(SSH_{model})) but it becomes very sensitive to the amount of instrumental noise (yellow spread delimiting a $\pm$ 50% increase/decrese noise level)

On the resolutions of ocean altimetry maps

Maxime Ballarotta[1], Clément Ubelmann[1], Marie-Isabelle Pujol[1], Guillaume Taburet[1], Florent Fournier[1], Jean-François Legeais[1], Yannice Faugère[1], Antoine Delepoulle[1], Dudley Chelton[2], Gérald Dibarboure[3], Nicolas Picot[3]

[1]Collecte Localisation Satellite, Ramonville-Saint-Agne, 31520, France
[2]College of Earth, Ocean and Atmospheric Sciences, Oregon State University, Corvallis, OR
[3]Centre National d'Études Spatiales, Toulouse, 31400, France

Correspondence to: M. Ballarotta (mballarotta@groupcls.com)

Abstract. The DUACS system produces sea level global and regional maps that serve oceanographic applications, climate forecasting centers, geophysics and biology communities. These maps are generated using an optimal interpolation method applied to altimeter observations. They are provided on a global ¼°x ¼° (longitude x latitude) and daily grid resolution framework (1/8°x1/8° longitude x latitude grid for the regional products) through the Copernicus Marine Environment Monitoring Service (CMEMS). Yet, the dynamical content of these maps does not have a full ¼° spatial and 1-day resolution, due to the filtering properties of the optimal interpolation. In the present study, we estimate the "effective" spatial and temporal resolutions of the newly reprocessed delayed-time DUACS maps (aka, DUACS-DT2018). Our approach is based on the ratio between the spectral content of the mapping error and the spectral content of independent true signals (along-track and tide gauge observations), also known as the Noise-to-Signal ratio. We found that the spatial resolution of the DUACS-DT2018 global maps based on sampling by three altimeters simultaneously ranges from ~100km-wavelength at high latitude to ~800km-wavelength in the Equatorial band and the mean temporal resolution is ~28 days period. The mean effective spatial resolution at mid-latitude is estimated to ~200km. The mean effective spatial resolution is ~130 km for the regional Mediterranean Sea and for the regional Black Sea products. An inter-comparison with previous DUACS reprocessing systems (aka, DUACS-DT2010 and DUACS-DT2014) highlights the progress of the system over the past 8 years, in particular a gain of resolution in highly turbulent regions. The same diagnostic applied to maps constructed with two altimeters and maps with three altimeters confirms a modest increase of resolving capabilities and accuracies in the DUACS maps with the number of missions.

1 Introduction

The Data Unification and Altimeter Combination System (DUACS) generates, as part of the CNES/SALP project and the Copernicus Marine Environment and Monitoring Service (CMEMS), delayed-time (DT) multi-mission altimeter Sea Level Anomaly (SLA) Level 3 (along-track cross-calibrated) and Level 4 (multiple sensors merged as maps or time series) products. A full reprocessing of these products is carried out approximately every 3 years and covers the period 1993 – now. The

reprocessing benefits from improvements associated with optimized mapping parameters and new altimeter corrections which are based on standards recommended for altimeter products by the different agencies and expert groups (Ocean Surface Topography Science Team (OSTST), the ESA Quality Working groups and the ESA Sea Level Climate Change Initiative project members). The previous reprocessing was released in 2014 (DUACS-DT2014, see Pujol et al., 2016) and the new release, namely DUACS-DT2018, is available since April 2018 (Taburet et al., 2019).

The Level 4 DUACS-DT global maps are constructed from optimal interpolation (Bretherton et al., 1976, Le Traon, 1998, Ducet et al., 2000) of Level 3 altimeter observations and are provided on a regular ¼°x ¼° longitude x latitude and daily grid resolution framework (1/8°x1/8° horizontal sampling for the regional Mediterranean and Black Sea products). However, the optimal interpolation used in DUACS does not allow the restitution of the full dynamical spectrum of the ocean, limiting the capability of retrieving small mesoscale in Level 4 products (Chelton et al, 2011 and 2014).

The "effective" resolution corresponds to the spatio-temporal scales of the features that can be properly resolved in the maps. The spatio-temporal resolution of the previous Level 4 global SLA products was estimated by Chelton et al. (2003, 2011, 2014) based on estimates of the mapping errors in Sea Surface Height (SSH) fields constructed from altimeter data, or spectral ratio analysis between maps and along-track altimeter data. Their analysis suggested mid-latitude spatial resolution capability of the observations ranging from ~2° to 6°, depending on the number of altimeters used in the merging and the sampling pattern of the ground track (~2° for tandem mission T/P-Jason 5 days offset between parallel tracks, 6° for T/P mono-mission merging). The temporal resolution capability of the observations for a tandem T/P-Jason mission was estimated ~20 days.

In the present study, we further investigate the effective resolution of the DUACS-DT gridded products using a spectral approach. The objective of the paper is threefold: 1) to deliver the spatial distribution of the effective resolutions as key information to the users about the quality and the limitations (in term of resolution) of the newly produced DUACS-DT2018 gridded products, 2) to access and compare the spatial and temporal resolution capabilities of the DUACS-DT2018, DUACS-DT2014 and DUACS-DT2010 maps (i.e., to identify the impact of system upgrades), and 3) to verify the impact of the varying satellite constellation on the effective resolutions of the maps. The paper is organized as follow: the data and method are introduced in section 2. In section 3, we present our results. Finally, a discussion and a conclusion are provided in section 4. A sensitivity study on the choice spectral criterion to estimate the resolution and a comparison of various approaches to estimate the resolution is given in the Appendix.

2 Data and method

2.1 Input data

In the present study, we consider two kinds of data:

- Independent dataset: we used two independent (i.e., not used in the mapping) datasets to evaluate the effective resolutions of the maps: 1) Level 3 CMEMS SLA from independent 1Hz along-track and 2) the SLA estimated from tide gauges. The along-track SLA are constructed using a procedure similar to Level 3 CMEMS products and are

used to estimate the effective spatial resolution. The SLA at tide gauge locations originate from the Global Sea Level Observing System and Climate and Ocean Variability, Predictability and Change (GLOSS-CLIVAR) network and is used to estimate the effective temporal resolution. GLOSS-CLIVAR data are worldwide and available with daily sampling.

- The maps of SLA are constructed using optimal interpolation, based on the a priori statistical knowledge of the field (e.g., variance, correlation scales, noise). The mapping procedure is based on merging of calibrated multi-satellite altimeter (Level 3) data and follows the same protocol as described by Pujol et al. (2016) for the DUACS DT2014. Taburet et al. (2019) give the full description and validation of the DUACS-DT2018 global and regional products. The main differences between the DUACS-DT2014 and the DUACS-DT2018 processing consist of an improved along-track processing (e.g., improved orbit correction, wet troposphere correction, ocean tide correction and a new mean sea surface) and updated a priori knowledge of the SLA variance and optimized selection of the data in the optimal interpolation. The maps tested here are computed specifically for this study in several constellation scenario, keeping at least 1 mission out to allow an independent assessment of the resolution. The DUACS-DT products, formerly known as AVISO products, are referenced in the CMEMS catalogue as "OCEAN GRIDDED L3/4 SEA SURFACE HEIGHTS AND DERIVED VARIABLES REPROCESSED" products.

2.2 Method

Our method to estimate the effective spatial resolution is based on the ratio between the spectral content of the mapping error and the spectral content of independent signal (along-track observations previously mentioned).

$$NSR(\lambda_s) = \frac{S_{diff}(\lambda_s)}{S_{obs}(\lambda_s)} \qquad (1)$$

Where, λ_s is the spatial wavelength, $S_{diff}(\lambda_s)$ is the power spectral density of the difference (SLA_{obs} - SLA_{map}), $S_{obs}(\lambda_s)$ is the spectral density of the independent observation.

The algorithm to compute the spatial effective resolution follows 4 main steps:

- A coastal editing is applied in a 100 km coastal band (only for the global products) to remove the increased errors in the coastal area.
- Gridded data are interpolated to the locations of the independent along-track data.
- Along-track and interpolated data are divided into overlapping 1500km long segments every 300 km for the global products (500km long segments for the Mediterranean Sea products and 300km long segments for the Black sea products). Each segment is saved in a database and referenced by it median (longitude, latitude) coordinates.
- Finally, between latitudes 90°S-90°S and longitudes 0-360°E, we consider 10°x10° longitude x latitude boxes for the global products (5°x5° longitude x latitude boxes for the Mediterranean Sea product, and 3°x3° longitude x latitude boxes for the Black Sea product) every 1° incremental step. All available segments referenced within the 10°x10°

box are selected to compute the power spectral densities based on the Welch method (1967). Prior to spectral computation, signals are detrended and we applied a Hanning window. The effective resolution is then given by the wavelength λ_s where the $NSR(\lambda_s)$ is 0.5.

The method (applied to the altimetry product) is illustrated in Figure 1 with the data selection and interpolation step to the spectral analysis. The total number of averaged segments in each 1°x1° longitude x latitude box is shown in Figure A1a) for the global product, Figure A1b) for the Mediterranean Sea product and the Black Sea product. Due to the coastal editing, the number of computed segments in the global product analysis is less than 1000 near the coast and ~1500 in the open ocean. In the Mediterranean Sea the number of segments is ~400 and ~250 for the Black Sea. A limitation of the present spectral approach is the need to rely on coastal for estimation of the resolution in the two regional products. It is worth noting, that we probably underestimate the resolution capability of the maps since we are estimating the spatial effective resolution of degraded maps to keep an independent dataset aside.

A comparison of SLA maps with independent tide gauge dataset is carried out to estimate the effective temporal resolution. The approach is like the estimate of the effective spatial resolution and based on the computation of the ratio between the spectral content of the mapping error and the spectral content of the true tide gauge signal (Eq. 2):

$$NSR(\lambda_t) = \frac{S_{diff}(\lambda_t)}{S_{obs}(\lambda_t)} \qquad (2)$$

Where, λ_t is the temporal wavelength, $S_{diff}(\lambda_t)$ is the power spectral density of the difference (SLA_{obs} - SLA_{map}), $S_{obs}(\lambda_t)$ is the spectral density of the independent observation.

We computed the effective temporal resolution from each tide gauge time series of the GLOSS-CLIVAR network. The temporal domain covers the period 19930101 - 20151231. The computation for each time series follows 3 main steps:

- At each tide gauge location, we extract the gridded SLA time series that is most highly correlated with tide gauge time series (note that the maximum distance separation of the grid point that is most highly correlated with each tide gauge is 100km on average and can be as large as 300km)
- Each highly correlated time series (based on correlation criterion > 0.8) is subsampled into 100-day segments to compute the spectral densities S_{diff} and S_{obs}. The length of each segment must be set when estimating power spectral density using Welch's method. By subsampling into 100-day segments, we limit the frequency range to only periods shorter than 100 days. We performed a sensitivity study on longer segment length (200 days and 300 days periods) and found similar global averaged effective temporal resolution (26 days for 300 days long segment, 27 days for 200 days long segment and 28 days for 100 days long segment). Note that consideration of time series longer than 100 days reduced the number of realizations because of occasional gaps in some of the data records. This can have an impact on the local estimation of the effective temporal resolution, e.g. from 100-day to 300-day segments we lose

some continuous time series that do not contain >=200-day segment. In these cases, the spectral analysis cannot be performed.

- The effective temporal resolution at each tide gauge location is given by the period λ_t where the ratio $NSR(\lambda_t)$ is equal to 0.5.

Note that this estimation of the temporal resolution is subject to an important caveat: the estimation is based mainly on coastal locations which may be contaminated by altimetry errors. Additionally, it may be not be fully representative of the temporal resolution of the DUACS maps which combine various oceanic regimes (e.g., coastal, offshore high variability, offshore low variability regimes). Our results may therefore be crude but useful estimates of the temporal resolution.

We somewhat subjectively define the effective resolutions to be the wavelength above which the NSR exceeds 0.5. In other words, it corresponds to the threshold where the mapping error variance is two times smaller than the observed true signal variance. The methodology used here is similar to that Chelton et al. (2018), except that Chelton et al. (2018) consider the NSR in the spatial domain, whereas we here consider the NSR in the wavenumber domain. To illustrate and discuss the impact of the choice of the NSR criterion on the resolution, a sensitivity study is provided in the Appendix B. We demonstrate that the resolution can be ~30% coarser with NSR = 0.25 (SNR = 4) and > 30 % coarser with a more conservative NSR criterion (e.g., SNR=10, as recommended by Chelton et al, 2018). It is worth mentioning that various approaches may exist to estimate the resolution (e.g., spectral magnitude ratio, filter transfer function). All measures of resolution have their advantages and drawbacks. We discuss in Appendix A the impact of using these different approaches in the estimation of the effective resolution.

3 Results

3.1 Effective resolutions of the DUACS-DT2018 maps

The effective spatial resolution of the DUACS-DT2018 global maps is shown in Figure 2a. Resolution was computed for maps constructed with three altimeters (Cryosat-2, HY-2, Jason-2) over the period 20140412-20151231 and Saral/Altika data were used as an independent dataset. We believe that this assessment of the spatial resolution based on maps constructed with three altimeter missions may be considered as a reasonable averaged estimate since ~3 altimeter missions are used in the merging for the CMEMS products 70% of the time over the period 19930101-20170515. The resolution ranges from ~100 km wavelength at high latitudes to ~800km wavelength near the Equator, with a mean resolution at mid-latitude near 200km. Considering that eddy radius characteristic can be estimated as 20-25% of the wavelength (Chelton et al., 2011; 2018), this means that ~25km radius structures are properly resolved in the maps at high latitudes, ~200km radius structures are resolved in the Equatorial band and ~50km radius structures are resolved at midlatitudes. The effective spatial resolution of the DUACS-DT2018 Mediterranean Sea maps ranges from 90 to 160 km wavelength (Figure 2b). The averaged resolution is ~130 km

wavelength over the basin. The effective spatial resolution of the DUACS-DT2018 Black Sea maps ranges from 100 to 150 km wavelength and the averaged resolution is ~130km wavelength over the basin (Figure 2b).

The effective temporal resolution of the DUACS-DT2018 maps ranges from 10 to 49 days period (Figure 3). The temporal resolution is heterogeneously distributed over the global ocean, particularly in the inter-tropical band where a wide range of scales are found, linked to the mixture of continental tide gauges and island tide gauges, with the latter being more representative of open-ocean conditions. At mid-to-high latitudes the temporal scales are between 14- and 28-day periods, coherent with the temporal correlation scales applied in the mapping process. The globally averaged effective temporal resolution is estimated ~28 days period.

The globally averaged resolutions of about 200 km by 28 days period are consistent with the resolutions reported by Chelton et al. (2011; 2014) and Pujol et al. (2016). Using the spectral ratio method (see Appendix B), they found spatial resolution slightly better than 200 km at mid-latitude in Pacific Ocean.

3.2 Evolution of the DUACS system

We here investigate the impact of the DUACS system upgrade from 2010 to 2018 to highlight the progress of the DUACS processing. Resolutions were computed for maps constructed with two altimeters (Topex-Poseidon and Jason1) over the period 20030101-20041231 and Geosat Follow On data were used as independent dataset. To identify the impact of the DUACS system upgrade, we computed the relative improvement/deterioration of the effective resolutions (expressed in percentage) for the upgrade DT2010 to DT2014, and DT2014 to DT2018 (Figure 4). Negative (positive) value means finer (coarser) resolution with the upgrade. The comparison of the DT2010 and DT2014 processing shows finer resolution (improvement > 2%) in DT2014 than in DT2010 in the high variability regions, e.g. the Gulfstream system, the Kuroshio system and the Antarctic Circumpolar Current (ACC) (Figure 4a). These improvements are associated with updated processing such as improved instrumental and atmospheric correction, tide correction, inter-calibration method and smaller correlation scale in the mapping process. Coarser resolutions in DT2014 than in DT2010 are found in the Equatorial band and are potentially linked to larger correlation scales applied in this region in the DT2014, as reported by Pujol et al. (2016). Although the DT2018 and DT2014 global maps have similar mean effective spatial resolution, regional investigation highlights ~2 to 10% improved resolution in DT2018 in highly turbulent regions (Figure 4b), such as the Equatorial region, the Gulfstream system, the Kuroshio system as well as some regions in the ACC. These improvements are linked to the new mapping standard (optimized selection of the observations in the turbulent region and a priori knowledge of the SLA variance based on a longer period in DT2018). The loss of resolution in the South Equatorial Atlantic is not understood yet.

Similar comparison is performed for the Mediterranean and Black Sea regional products focusing on the upgrade DT2014 to DT2018. Resolutions were computed for regional DUACS maps constructed with three altimeters (Jason-2, Cryosat-2, HY-2) over the period 20140412-20151231 and Saral/Altika was used as an independent dataset. The resolution capability of the Mediterranean Sea maps is slightly finer (~4%) in DT2018 than in DT2014 (Figure 4c). The largest improvements (>6%) are found the western Mediterranean basin. The resolution in DT2018 is slightly coarser in the closed seas (Adriatic Sea and

Aegean Sea). In these regions, the limited number of along-track data restricts a reliable interpretation of the spectral signal (see Figure A1). The resolution capability of the Black Sea maps is on average slightly finer (~3%) in DT2018 than in DT2014, although a deterioration is found in the central part of the basin, which is also linked to a reduced number of spectral computations (Figure 4c).

The DUACS-DT2018 and DUACS-DT2014 maps have similar mean effective temporal resolution of ~ 28 days period (globally averaged difference < 4%). The differences can be locally larger than 15%, near 30° along the Japanese coast (~10 days gain), as shown in Figure 5. In these regions, the temporal resolution in the DUACS-DT2018 is finer than in DUACS-DT2014. These regions also coincide with the largest increased correlation score in the DUACS-DT2018 between SLA time series from maps and from independent tide gauge sensors (Taburet et al., 2019). These coastal improvements are linked to the new altimeter standards in coastal regions in DT2018 (Taburet et al., 2019).

3.3 Impact of altimeter constellation on the effective spatial resolution

Since the number of altimeter data processed by the DUACS system varies with time (according to the availability of satellites and the data quality), we investigated the impact of the constellation on the effective spatial resolution. Figure 6 illustrates the impact of the number of altimeters (2 or 3 missions) used in the mapping on the effective spatial resolution. We verify, with our diagnostic, modest increases of resolving capabilities in the DUACS maps with increasing number of altimeters and found a globally averaged gain of resolution of ~5% from maps constructed with three altimeters compared with two altimeters. The resolution improvements may be considered as modest. The reason is that the same covariance parameters are used in the optimal interpolation (OI) procedure, regardless of how many altimeters are available. These OI parameters exert very strong constraints on the filter transfer function of the OI procedure.

Regional gains of resolution can be larger than 10%. Additionally, it is possible to identify the improved resolving capability when a new mission is introduced in the DUACS system: for example, Figure 6a illustrates the improved resolving capability when mission HY-2 is introduced in the mapping, Figure 6b illustrates the improved resolving capability when mission Cryosat-2 is introduced in the mapping. It is shown that the major contribution of the HY-2 mission in the mapping is in the high variability regions (Gulfstream, Kuroshio, Agulhas systems) while Cryosat-2 contributes in the mid-to-high latitude regions. On the global scale, the distribution of the effective spatial resolution is shifted toward shorter scales when the number of missions used in the merging increases (Figure 7) or when recent altimeters are used in the interpolation (e.g., compare the resolution maps from DT2018 constructed with historical Jason-1/Envisat versus the maps from DT2018 constructed with currently operational missions Jason2/HY-2 or Jason-2/Cryosat2).

4 Discussion and conclusions

The present study investigates the resolving capability of the DUACS delayed-time gridded products (Global, Mediterranean Sea and Black Sea) delivered through the CMEMS catalogue. The key results are summarized in Table 1. Our method is based

on the Noise-to-Signal spectral ratio to estimate the resolution. While along-track altimeter data resolve wavelength scales in the order of few tens of kilometers (Dussurget et al., 2011, Dufau et al. 2016), we found that the merging of these along-track data into continuous maps in time and space leads to properly resolved structures with a wavelength scale of 100 km (a feature radius scale of ~25 km) at high latitudes to 800 km (a feature radius scale of ~200 km) near the Equator in the global gridded product and with a temporal scale of about 28-day period. The same analysis applied to the regional Mediterranean Sea and Black Sea products showed resolving capability of structure with feature radius resolution of ~30 km, which corresponds to ~3 grid spacing.

These results are consistent with previous investigations. Based on a spectral ratio approach (cf. Appendix B), Chelton et al. (2011) estimated a wavelength resolution of ~200 km for DT2010 and Chelton et al. (2014) estimated a wavelength resolution of ~180 km for DT-2014 in the mid-latitude Pacific Ocean. Our analysis based on the DT2018 global maps, suggests a wavelength resolution of ~200 km at mid-latitudes. As illustrated in Figure 8, we verified our estimation of the zonally averaged feature radius resolution of the mesoscale structures that can be properly mapped is smaller than the eddy scales computed by Chelton et al. (2011). The eddy length scales range from ~70 km at high latitudes to ~180km near the Equator. The effective resolution is ~1.6 smaller than the eddy length scale. Additionally, we confirm that the minimum 4 weeks lifetime criteria used by Chelton et al. (2011) to identify and follow eddies seems to be compatible with the 28-day resolution capability. Note that our time scale estimation is based mainly on coastal locations and might not be representative of all oceanic regime.

The comparison of the DUACS-DT2018 reprocessing with former DUACS reprocessing (DT2010 and DT2014) reveals that finer structures are mapped in the global and regional Mediterranean Sea DT2018 products. For the Black Sea product, the interpretation is more complex due to the small dimension of the basin and the limited amount of spectral computation. Globally, we found that the largest improvements reach 20% and are mainly in high variability regions, associated with the new mapping standard (e.g., optimized selection of the along-track data, new a priori knowledge of the signal variance based on 25 years of altimetry data, updated correlation scales for the regional Mediterranean Sea product) and new altimeter standards (e.g., instrumental and atmospheric corrections, tide corrections, inter-calibration method). The improvement patterns between DT2014 and DT2010 global maps is similar to those found by Pujol et al. (2016) using statistical comparison between maps and independent along-track, and drifters' datasets. The improvement patterns between DT2018 and DT2014 global maps coincide with those found by Taburet el al. (2019) for the validation of the DT2018 products. Using statistical comparison between maps and independent along-track altimeter data, Taburet et al. (2019) also found improvement (~3-4%) of the mapped mesoscale structures, in the high variability region and in the western Mediterranean Sea basin. Note that, at global scale, Taburet et al. (2019) diagnosed the largest improvements between DT2018 and DT2014 in the coastal regions which are partially edited (along 100km coastal band) in the processing for estimating the spatial scale in this study, but they are detected with the temporal scale analysis, showing shorter timescale in the DT2018 compared with DT2014.

Several studies showed that at least two altimeters are required to accurately map the SSH mesoscale structures (Le Traon and Dibarboure 1999; Ducet et al. 2000; Pujol and Larnicol 2005; Dibarboure et al. 2011; Chelton et al. 2007; 2011) and up to four altimeters are required for Near-Real-Time products (Pascual et al. 2006), because only past observation are available for the

mapping. This reduced number of observations has an impact on the estimation of the sea surface height. The present study reinforces these findings, showing that the resolution capability increased ~10-20% at regional scale from the merging of data from two to three altimeters.

It is worth noting, that we probably underestimate the resolution capability of the maps since we are estimating the spatial effective resolution of degraded maps to keep an independent dataset aside. The resolution might hence be somewhat finer in the distributed CMEMS products. Although the satellite constellation ranges from 1 to 5 altimeter(s) between 19930101 and 20170715, we believe that our estimation of the spatial resolution based on maps constructed with three altimeter missions may be considered as a reasonable averaged estimate since ~3 altimeter missions are used in the merging for the CMEMS products 70% of the time over the period 19930101-20170515. We can expect > 5% finer resolution over period where more than 4 altimeters are available (i.e., recent period). Likewise, we can expect on average 5% coarser resolution when only 2 altimeters are available.

To conclude, the number and the quality of altimeter simultaneously operational, the along-track configuration and sampling pattern, the weight given to the altimeter data in the mapping procedure and the choice of threshold SNR are key factors controlling the resolution capability of the DUACS gridded products. One may expect that, in permitting to observe finer mesoscale/sub-mesoscale structures (Dufau et al., 2016, Pujol et al, 2012), future instrumental systems based on large-swath altimeter (such as Surface Ocean and Water Topography (SWOT)) combined with new mapping technique based on dynamic interpolation (Ubelmann et al., 2016), will push the maps resolution toward new limit.

Code and data availability. The DUACS system source code is not publicly available. The code for the spectral analysis is released under GNU General Public License v3.0 and is available at https://github.com/mballaro/scuba. DUACS all satellites gridded data and along-track data are available through the CMEMS website: http://marine.copernicus.eu/. Specific maps used in our study are based on merging of three or two satellites and are available on request by contacting M. Ballarotta (mballarotta@groupcls.com).

Acknowledgment

This work is a contribution to the CMEMS R&D activities, the CNES-SALP project and the BOOST-SWOT project funded by ANR (project number ANR-17-CE01-0009-01). We would like to thank Lee Lueng Fu for his suggestions on the manuscript, Tom Farrar for his detailed and inspiring review and one anonymous referee. All their comments significantly added value to this paper.

References

Bretherton, F.P., Davis, R.E., and C. Fandry: A technique for objective analysis and design of oceanographic instruments applied to MODE-73, *Deep-Sea Res.*, 23, 559–582,10 doi:10.1016/0011-7471(76)90001-2, 1976

Chelton, D. B. & Schlax, M. G. The accuracies of smoothed sea surface height fields constructed from tandem satellite altimeter datasets. *J. Atmos. Oceanic Technol.* **20**, 1276–1302 (2003)

Chelton, D. B., Schlax, M. G., Samelson, R. M.: Global observations of nonlinear mesoscale eddies, *Prog. Oceanogr.*, 91, 167–216, doi:10.1016/j.pocean.2011.01.002, 2011.

Chelton D., G. Dibarboure , M.-I. Pujol, G. Taburet , M. G. Schlax: The Spatial Resolution of AVISO Gridded Sea Surface Height Fields, OSTST Lake Constance, Germany, October, 28-31 2014, available at http://meetings.aviso.altimetry.fr/fileadmin/user_upload/tx_ausyclsseminar/files/29Red0900-1_OSTST_Chelton.pdf, 2014

Chelton, D., Schlax, M. G., Samelson, R. M., Farrar J.T., Molemaker M.J., McWilliams J.C., Gula J. : Prospects for Future Satellite Estimation of Small-Scale Variability of Ocean Surface Velocity and Vorticity, *accepted for publication in Progress in Oceanography*, 2018

Dibarboure, G., Pujol, M.-I. , Briol, F., Le Traon, P. Y. , Larnicol, G., Picot, N., Mertz F. & M. Ablain (2011) Jason-2 in DUACS: Updated System Description, First Tandem Results and Impact on Processing and Products, Marine Geodesy, 34:3-4, 214-241, DOI: 10.1080/01490419.2011.584826

Ducet, N., Le Traon, P-Y, and Reverdin G.: Global high-resolution mapping of ocean circulation from TOPEX/Poseidon and ERS-1 and-2, *Journal Of Geophysical Research-oceans* , 105(C8), 19477-19498, doi:10.1029/2000JC900063, 2000

Dufau,C., Orsztynowicz, M., Dibarboure, G., Morrow, R. and P.-Y. Le Traon: Mesoscale resolution capability of altimetry: present and future, *J. Geophys. Res.*, Oceans, 121 (7), pp. 4910-4927, doi: 10.1002/2015JC010904, 2016

Dussurget, R., Birol, F., Morrow, R. and P. De Mey: Fine Resolution Altimetry Data for a Regional Application in the Bay of Biscay, *Marine Geodesy*, 34:3-4, 447-476, doi: 10.1080/01490419.2011.584835, 2011

Le Traon, P.-Y., Faugère, Y., Hernandez, F., Dorandeu, J., Mertz, F. and M. Ablain: Can we merge GEOSAT Follow-On with TOPEX/POSEIDON and ERS-2 for an improved description of the ocean circulation?, *J. Atmos. Oceanic Technol*., 20, 889 – 895, doi: 10.1175/1520-0426(2003)020<0889:CWMGFW>2.0.CO;2, 2003

Le Traon, P. Y., Nadal, F., and Ducet, N.: An improved mapping method of multisatellite altimeter data, J. Atmos. Ocean. Tech., 15, 522–533, 1998.

Le Traon, P.Y.; Dibarboure, G. Mesoscale mapping capabilities from multiple altimeter missions. *J. Atmos. Ocean. Technol.*, 16, 1208–1223, 1999

Le Traon, P-Y: A method for optimal analysis of fields with spatially-variable mean. *Journal Of Geophysical Research-oceans*, 95(C8), 13543-13547, doi: 10.1029/JC095iC08p13543, 1990

Pujol, M. I., & Larnicol, G. 2005. Mediterranean Sea eddy kinetic energy variability from 11 years of altimetric data. Journal of marine systems, 58 (3-4), 121–142

Pujol, M.-I., Dibarboure G., Le Traon P.-Y., Klin P.: Using high-resolution altimetry to observe mesoscale signals, *J Atmos Ocean Technol* ., 29(9):1409–141, doi: 10.1175/JTECH-D-12-00032.1, 2012

Taburet, G., Sanchez-Roman, A., Ballarotta, M., Pujol, M.-I., Legeais, J.-F., Fournier, F., Faugere, Y., and Dibarboure, G.: DUACS DT-2018: 25 years of reprocessed sea level altimeter products, Ocean Sci. Discuss., https://doi.org/10.5194/os-2018-150, in review, 2019.

Smith, W.H.F. : Resolution of Seamount Geoid Anomalies Achieved by the SARAL/AltiKa and Envisat RA2 Satellite Radar Altimeters, *Marine Geodesy*, 38,644-671, doi: 10.1080/01490419.2015.1014950, 2015

Stammer, D.: Global characteristics of ocean variability estimated from regional TOPEX/POSEIDON altimeter measurements, Journal of Physical Oceanography, 27, 1743–1769, 1997

Ubelmann, C., Cornuelle, B., and Fu, L.-L.: Dynamic Mapping of Along-Track Ocean Altimetry: Method and Performance from Observing System Simulation Experiments, Journal of Atmospheric and Oceanic Technology, 33, 1691–1699, doi:10.1175/JTECH-D-15-0163.1, http://dx.doi.org/10.1175/JTECH-D-15-0163.1, 2016

Yale, M.M., Sandwell, D.T., and W.H.F. Smith: Comparison of along-track resolution of stacked Geosat, ERS-1 and Topex satellite altimeters, *Journal of Geophysical Research*, 100 (B8):15117–15127, doi: 10.1029/95JB01308, 1995

Welch, P.: The use of the fast Fourier transform for the estimation of power spectra: A method based on time averaging over short, modified periodograms, *IEEE Trans. Audio Electroacoust*., vol. 15, pp. 70-73, 1967.

Appendix A: other approach to estimate the resolution

1) Spectral magnitude ratio

Chelton et al. (2011, 2014) estimated the resolution of the DUACS DT2014 maps based on the calculation of the spectral magnitude ratio between the reference Stammer (1997) along-track spectrum and gridded SSH spectra. Similarly, we here estimate the resolution based on the spectral ratio between independent along-track and gridded SSH signals (Eq. 3). It is defined as follows:

$$SR(\lambda_s) = \frac{S_{map}(\lambda_s)}{S_{obs}(\lambda_s)} \quad (3)$$

, where $S_{obs}(\lambda_s)$ denote the power spectral density of the independent SSH along-track, $S_{map}(\lambda_s)$ denotes the power spectral density of the SSH map interpolated onto the independent along-track segment, SR the spectral ratio and λ_s the wavelength. The resolution is given by the wavelength λ_s where the spectral ratio is equal to 0.5 and is based on the conventional notion of a filter being characterized by its half-power filter cutoff wavelength (Chelton et al., 2011). To differentiate it from the effective resolution we named it "useful" resolution: "useful" for verifying the available and realistic amount of energy at a specific wavelength between two signals without considering their phase (e.g., useful for model sensitivity study).

Chelton et al. (2011, 2014) estimated the resolution of the DUACS-DT2010 and DUACS-DT2014 as the wavenumber at which the power is a factor of 2 smaller than the Stammer (1997) spectrum. From their analysis, they estimated spatial resolution of ~2° for the DUACS-DT2010, ~1.7° for the DUACS-2014. Chelton et al. (2014) found essentially the same resolution between maps constructed with 2 satellites or 4 satellites.

The resolution estimated with the SR method is shown in Figure A2a and the difference between the effective and useful resolution is shown in Figure A2b. The useful resolution of the DUACS-DT2018 maps ranges from 100km at high latitude to 500km near the Equator. The ratio effective/useful resolution suggests somewhat finer resolution in the intertropical band using SR approach and somewhat finer resolution at high latitude with the NSR approach. In other words, the amplitude of the mapped SSH spectral content is better in the inter-tropical band than the phase, whereas it is the opposite at high latitude. This feature highlights the difficulty to properly map propagating equatorial waves in the DUACS system. The two methods are equivalent at mid-latitudes.

2) Transfer function

The transfer function (H) measures the filtering properties of a system (Eq. 4). It is defined as follows:

$$H(\lambda_s) = \frac{CS_{obs-map}(\lambda_s)}{S_{obs}(\lambda_s)} \quad (4)$$

where $CS_{obs-map}(\lambda_s)$ is the cross-spectral density between the along-track data and the map interpolated onto the along-track segment, $S_{obs}(\lambda_s)$ is the power spectral density of the along-track. Note that in this case, the along-track is considered as non-independent (i.e., it is used in the mapping system). The resolution is given by the wavelength λ_s where H is equal to 0.5. It is the same definition used by Chelton and Schlax (1993) to estimate the resolution capability of an arbitrarily sampled dataset.

The resolution estimated with the transfer function method is shown in Figure A3a) and the difference between the effective resolution and the transfer function resolution is shown in Figure A3b). The transfer function resolution of the DUACS-DT2018 maps ranges from 100km at high latitude to 400km near the Equator. The difference between effective resolution versus transfer function resolution suggests somewhat finer resolution using the transfer function. This is directly linked to the fact that the along-track is here non independent. The assessment is undertaken below the track that is used in the filtering system. This diagnostic gives the filtering property of the system but "suffers" from the non-independency of the along-track dataset. The resolution may be different off-track.

These methods share the same number of spectrum calculation and number of segments used in the calculation (see Fig. A1) and each method has advantages and drawbacks. The spectral magnitude ratio compares the amplitude of the signals and the transfer function estimates the filtering properties from assimilated along-track. The function $NSR(\lambda_s)$, $SR(\lambda_s)$, and $H(\lambda_s)$ are shown for location (330°N, 45°N) in Figure A5. At this location, each function has a transition between 100km and 200km wavelength, separating the high mapping error regime for wavelength < 100km to the low mapping error for wavelength > 200km (Fig. A5a), separating the high amplitude error regime for wavelength < 100km to the low amplitude error for wavelength > 200km (Fig. A5b), and separating the filter regimes (Fig A5c).

Appendix B: Sensitivity to the Noise-to-Signal Ratio (SNR) criterion

We here investigate and discuss the impact of the NSR criterion on the estimation of the effective resolution. NSR criterion is used to define the resolution limit of the map. In the present study, we choose the NSR=0.5 criterion to define the resolution limit. This value may be considered too generous; therefore, we present below the effective resolution for three cases, motivated by the analysis performed in the spatial domain by Chelton et al. (2018):

- criterion of NSR=0.5
- criterion of NSR=0.25
- criterion of NSR=0.1

Figure B1a) represents the effective resolution using NSR=0.5 (SNR=2) criterion, Figure B1b) using NSR=0.25 (SNR=4) criterion and Figure B1c) NSR=0.1 (SNR=10). For each panel the resolution becomes finer poleward. The white areas correspond to the regions where the NSR threshold criterion is not achieved. These areas become larger in the inter-tropical region as well as at high latitudes when the NSR criterion decreases. For NSR=0.1, the resolution in the intertropical band cannot be computed with the method, since the NSR is above 0.1 for all scales. To further illustrate this, we show an example of NSR at one specific point (lon=346°E, lat=16°S) in Figure B2. The analysis shows that the NSR is greater than 0.3 (i.e., SNR < 3) in this location. This large-scale low coherency between maps and along-track may be linked to the misrepresentation

of the large scale and rapid equatorial waves (e.g., equatorial gravity waves) in the mapping process, which are filtered in the mapping process.

Despite the areas of missing values in Figures B1b) and B1c), we quantify the difference of effective resolution between criterion NSR=0.4 and NSR=0.25 (Figure B3) and NSR=0.5 and NSR=0.1. The difference between effective resolution computed with NSR=0.5 vs NSR=0.25 is < 30% (~60km) at mid-latitude and <=50% (400km) in the inter-tropical band. The difference between effective resolution computed with NSR=0.5 vs NSR=0.1 is < 50% (~100km) at mid-latitude and >50% in the inter-tropical band.

In conclusion, we here demonstrate that the choice of the NSR criterion has an impact on the estimation of the resolution. Setting more conservative criterion NSR=0.25 leads to ~30% coarser effective resolution. The strongly conservative criterion NSR=0.1 also reveals one of the major caveats in the DUACS maps processing: the poor representation of the large and rapid scale equatorial circulation. This issue should be addressed in the future version DUACS maps.

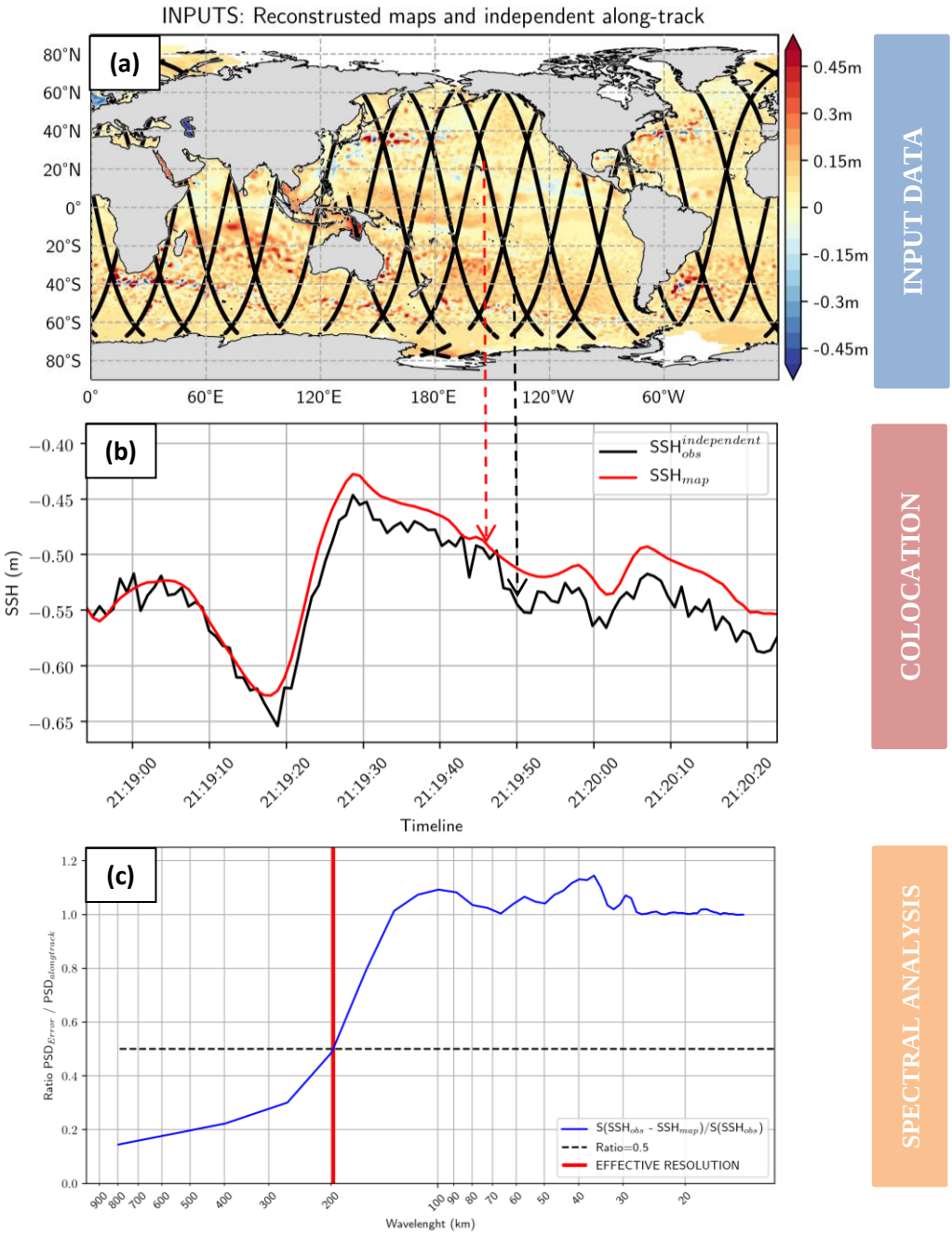


Figure 1: Schematic illustration of the methodology: a) input data selection, b) colocation SLA and gridded SLA and c) spectral analysis showing the ratio error spectrum to signal spectrum

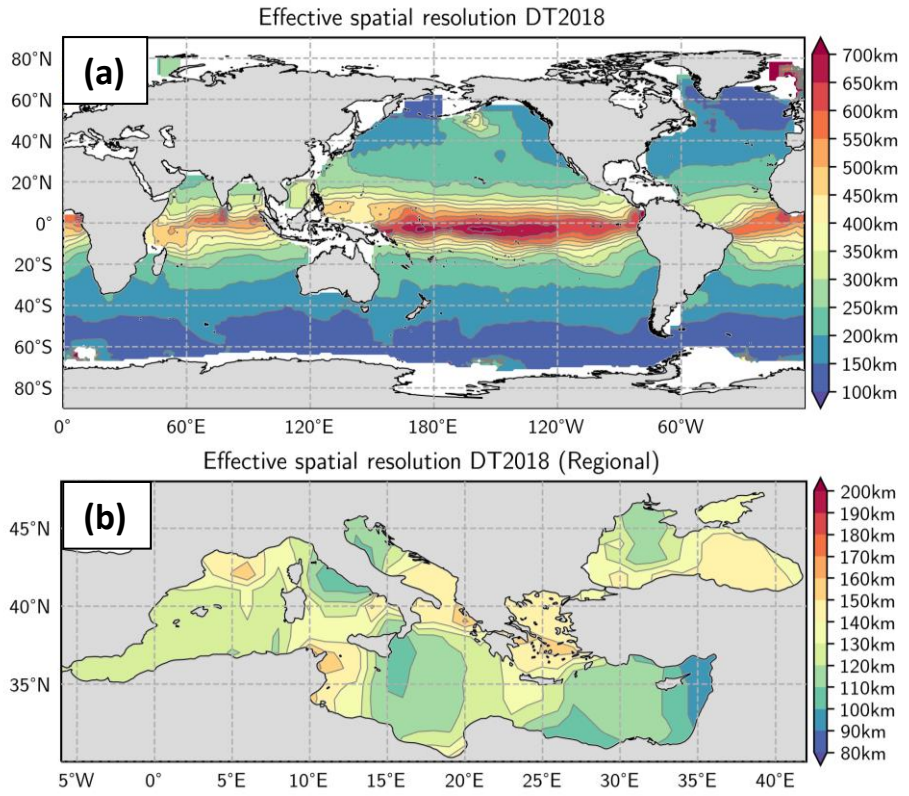


Figure 2: Effective spatial resolution in km of the DUACS-DT2018 maps for a) the Global Ocean product, b) the Mediterranean Sea and the Black Sea products. Unit in km

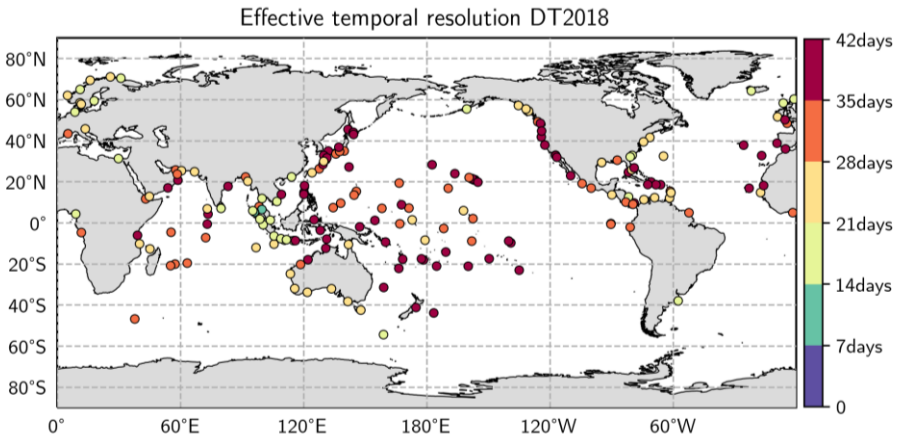


Figure 3: Effective temporal resolution in days of the DUACS-DT2018 maps. Unit in days

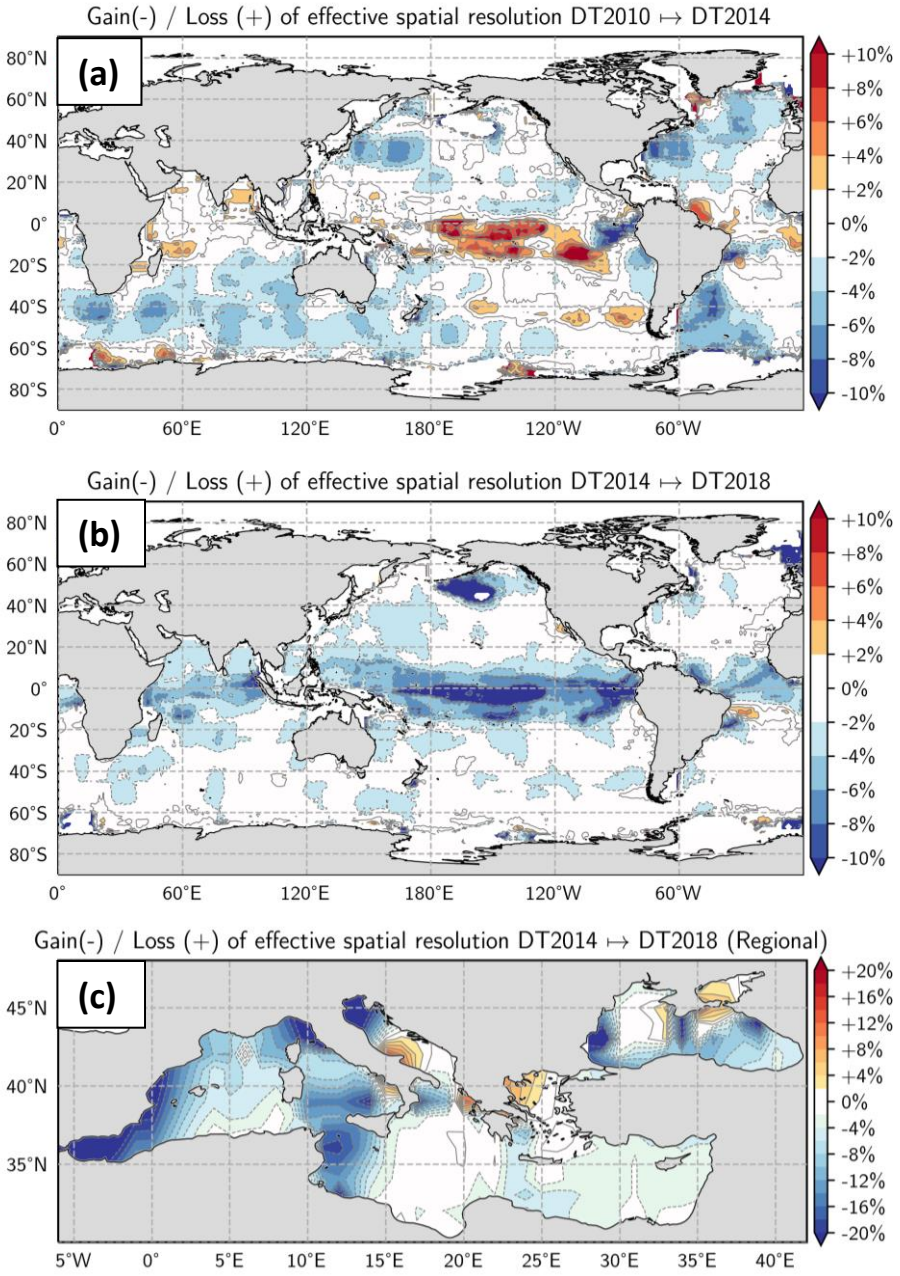


Figure 4: Gain/loss of effective spatial resolution for a) the Global Ocean product between DT2014 and DT2010, b) the Global Ocean product between DT2018 and DT2014, c) the Mediterranean Sea product and the Black Sea products between DT2018 and DT2014. Negative value means that the resolution capability is finer. Note the different colorbar scale between global and regional products

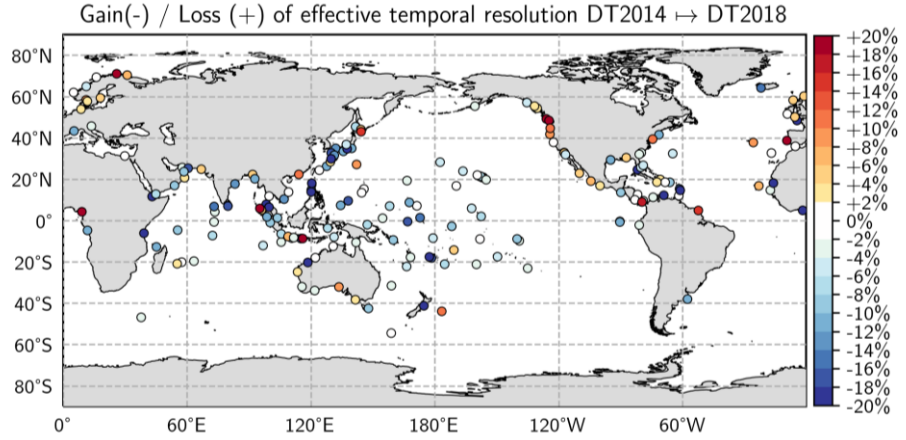


Figure 5: Gain/loss of effective temporal resolution between DT2018 and DT2014. Negative value means that the resolution capability is better in DT2018 than DT2014

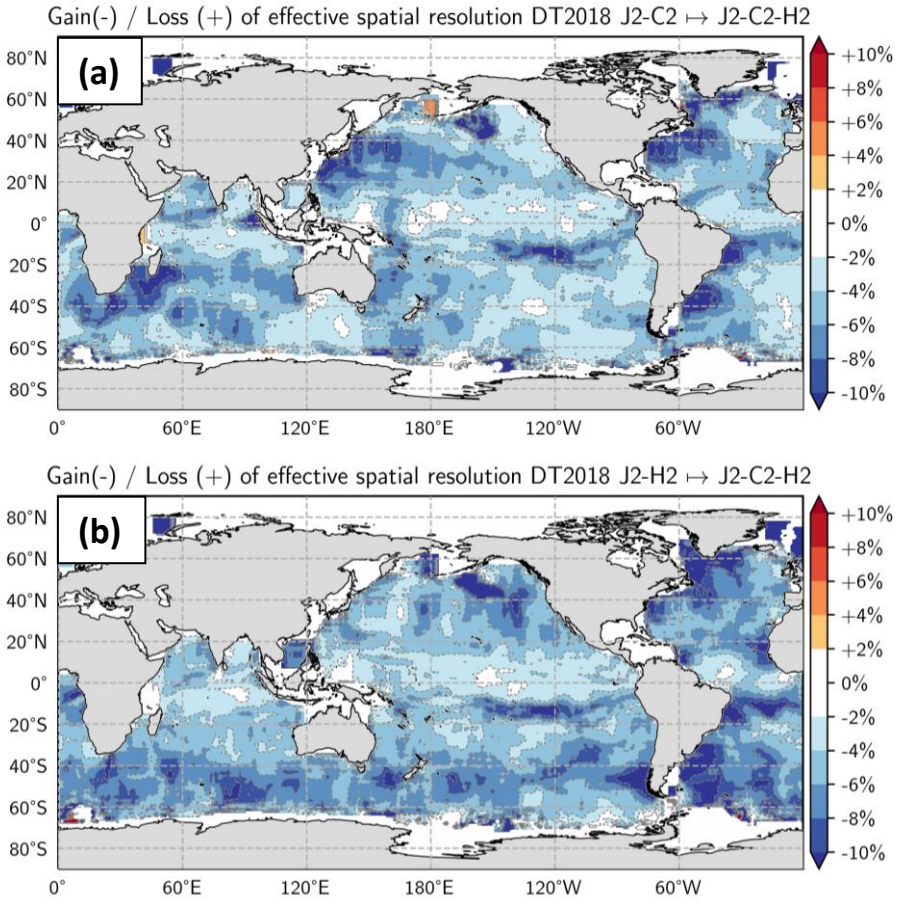


Figure 6: Impact of the satellite constellation on the effective resolution – Ratio of effective resolution of a) maps constructed with C2-H2-J2 vs C2-J2, and b) maps constructed with C2-H2-J2 vs H2-J2

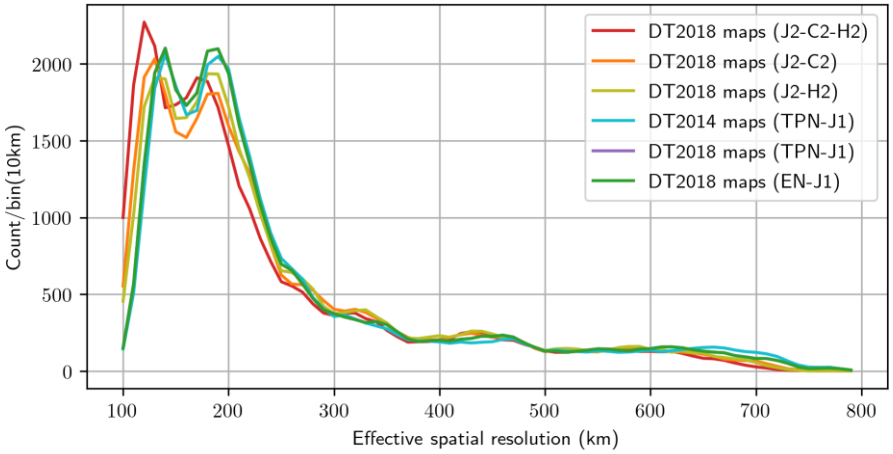


Figure 7: Distribution of the effective spatial resolution for various altimeter merging configuration

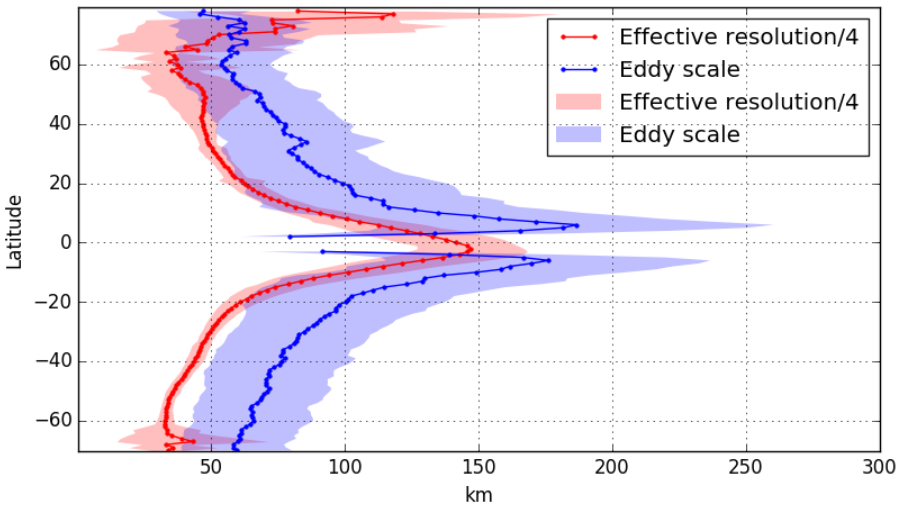


Figure 8: Zonally averaged eddy scale Ls (as in Chelton et al., 2011; and computed from the DUACS-DT2018 two satellites maps) and feature radius resolution of the mesoscale structures that can be properly mapped in DUACS (i.e., derived as 0.25xeffective resolution). Units in km

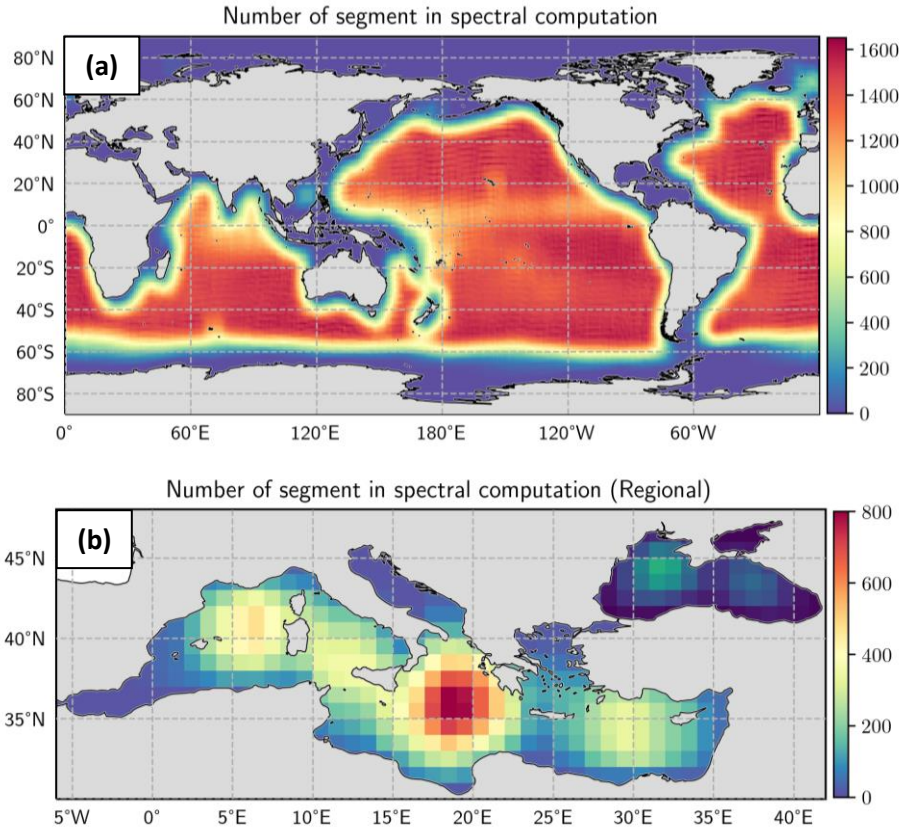


Figure A1: Number of segments used in the spectral computation for a) the global product and b) the regional Mediterranean and Black Sea products

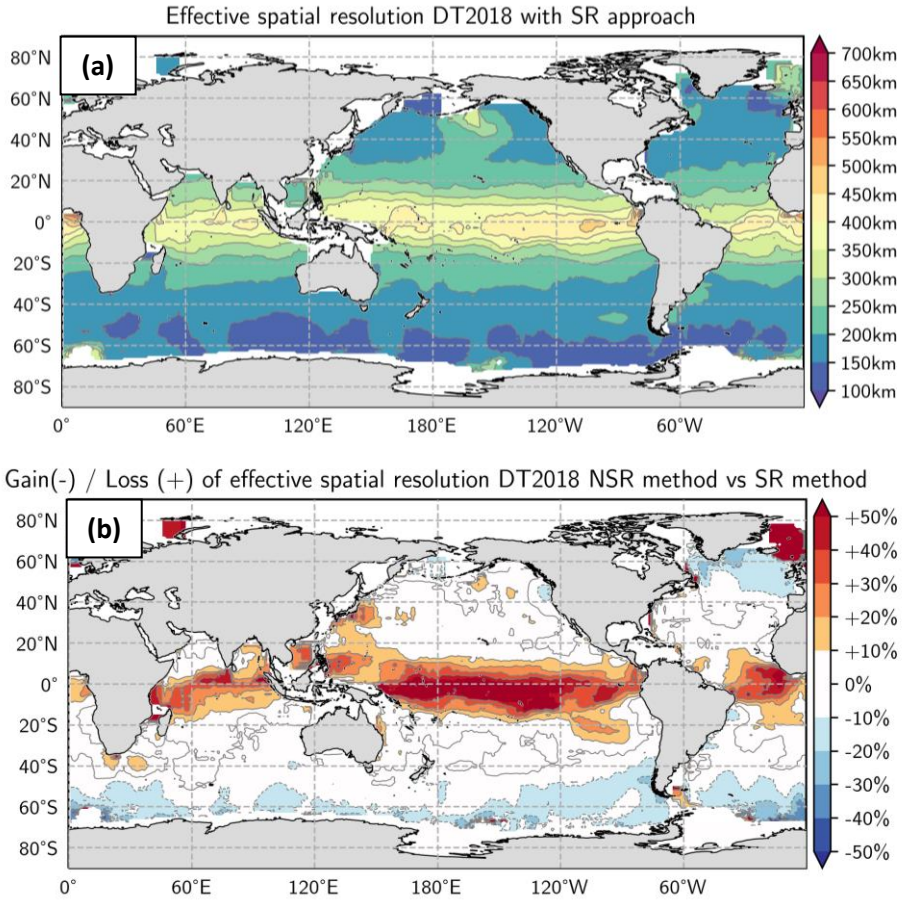


Figure A2: a) DUACS-DT2018 Useful resolution derived from spectral ratio approach and b) ratio effective / useful resolution for the DT2018 maps. Blue means finer resolution with NSR

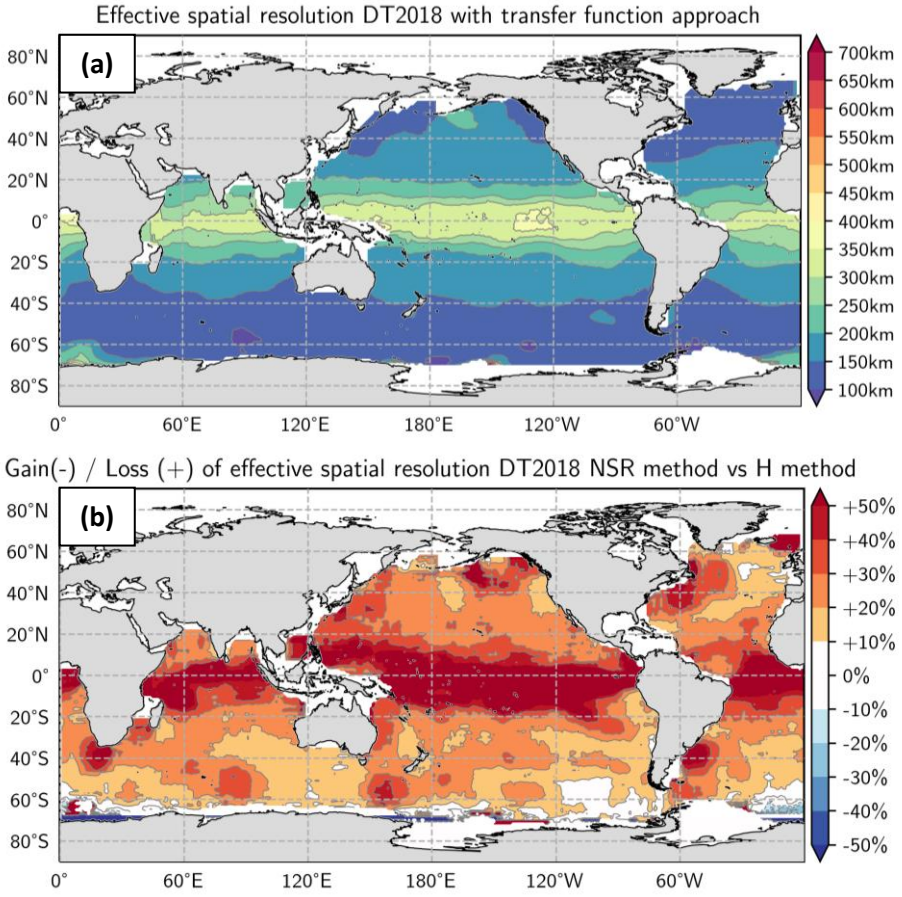


Figure A3: a) DUACS-DT2018 resolution derived from the transfer function approach and b) ratio effective / transfer function resolution for the DT2018 maps. Blue means finer resolution with NSR

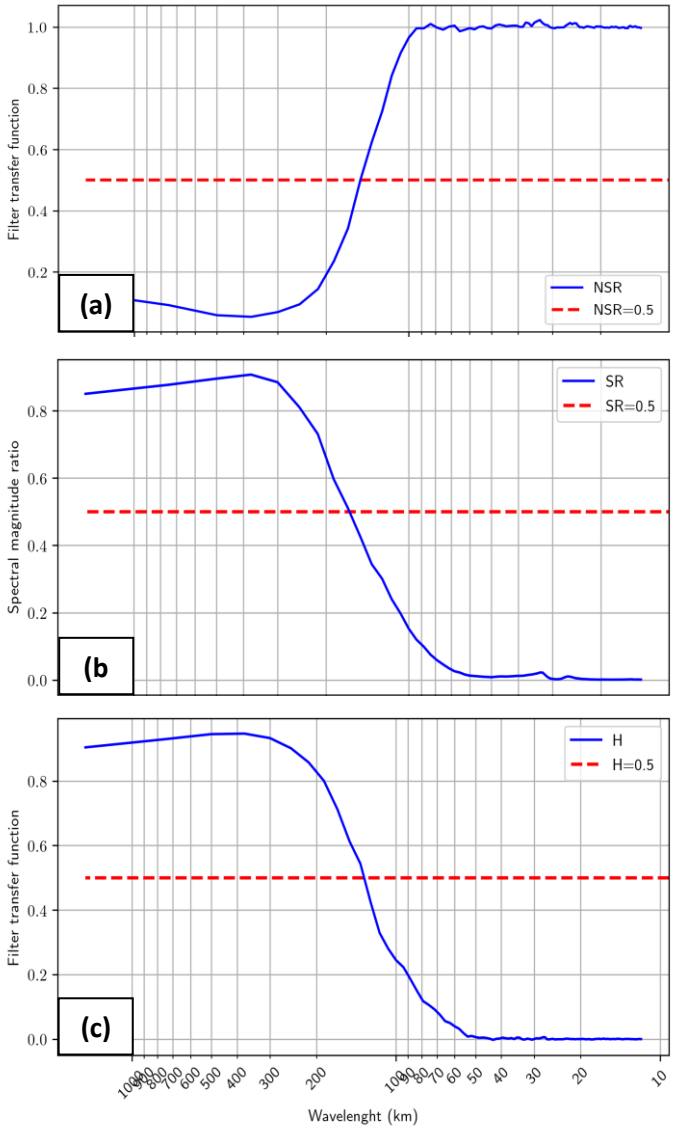


Figure A5: Illustration of the various spectral function used to estimate the resolution a 330°E, 45°N: a) with NSR method, b) with SR method, and c) with the transfer function

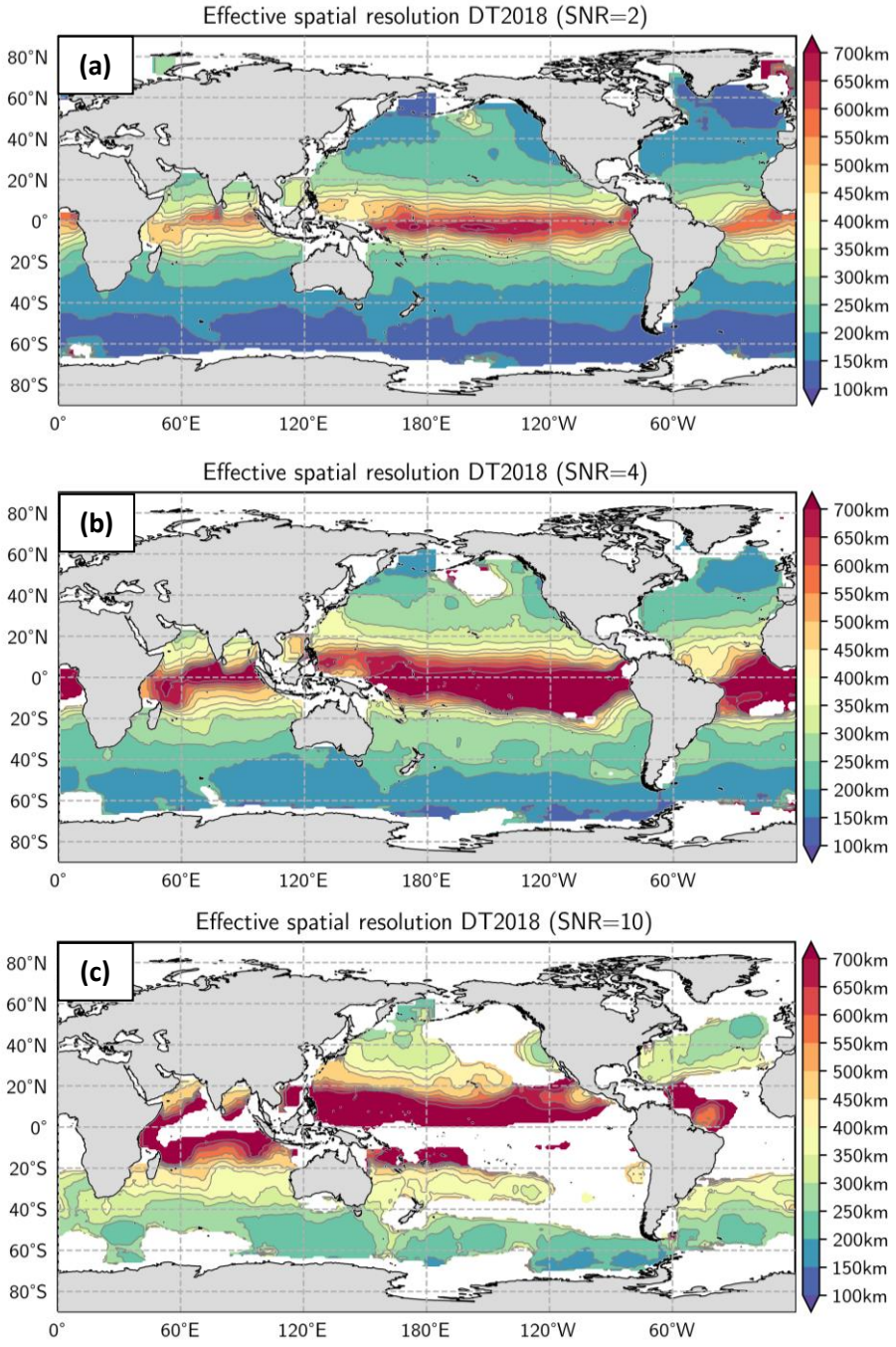


Figure B1: Effective resolution computed for three different SNR criteria: a) SNR=2, b) SNR=4 and c) SNR=10

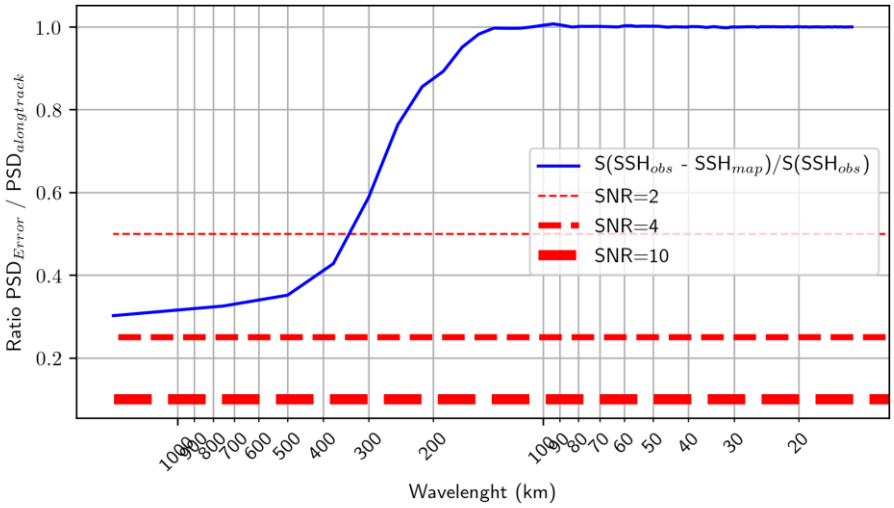


Figure B2: Ratio NSR at longitude 346°E and latitude 16°S. We illustrate that the ratio at this location is always above > 0.3 and so the resolution cannot be computed for SNR criterion > 4 (i.e., NSR < 0.25)

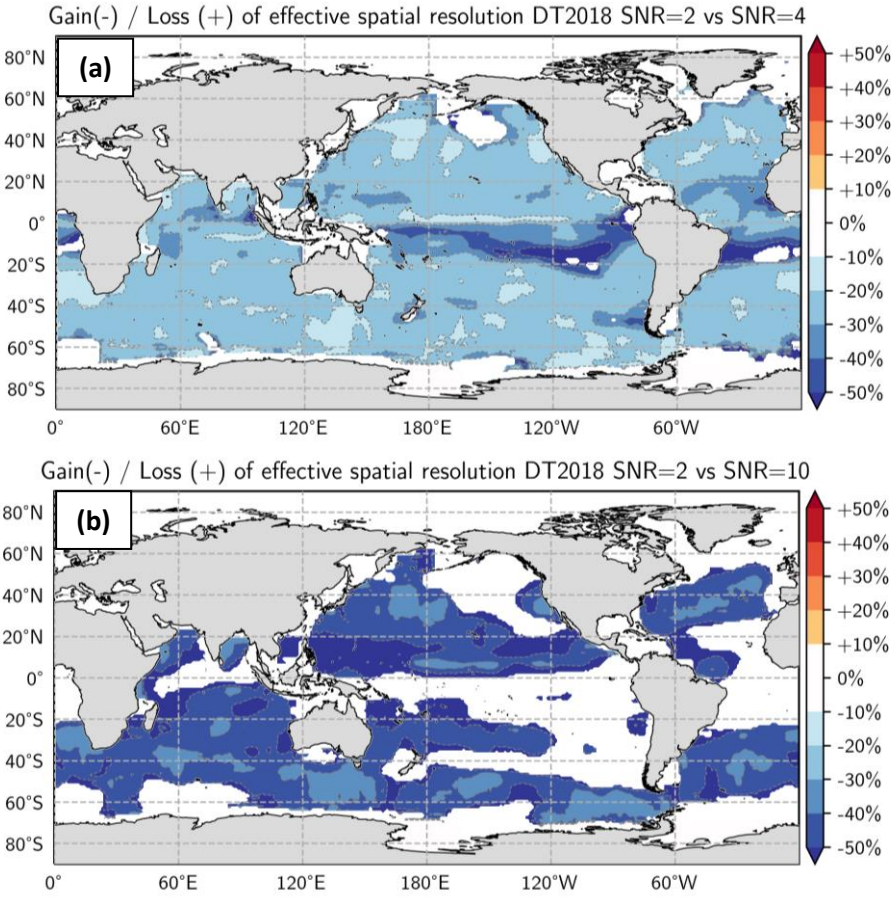


Figure B3: Ratio effective resolution computed with a) SNR=2 versus SNR=4 criterion and b) SNR=2 versus SNR=10 criterion. Blue means finer resolution with SNR=2

Table 1: Summary of the DUACS products spatial and temporal resolutions. (1) Not estimated due to the limited amount of tide gauges in the Mediterranean Sea and Black Sea

	SPATIAL FEATURE		TEMPORAL FEATURE	
Product	Effective resolution	Grid spacing	Effective resolution	Grid spacing
GLOBAL	100 to 800 km	4 to 30 km	~28 days	1 day
MED-SEA	~130 km	~10km	*(1)*	1 day
BLACK-SEA	~130 km	~10km	*(1)*	1 day