EXERCISE 7: Reduced-Rank Array Modes

Introduction

As described in Lecture 7, the 4D-Var state-vector increments $\delta\mathbf{x}_{\mathbf{a}}(t)$ can be expressed as a weighted sum of the array modes:

$$\delta\mathbf{x}_{\mathbf{a}}(t) = \sum_{i=1}^{M} \lambda_i^{-1}\left(\hat{\mathbf{w}}_i^{\mathrm{T}}\mathbf{R}^{-1/2}\mathbf{d}\right)\mathbf{\Psi}_i(t)$$

where $\left(\lambda_i, \hat{\mathbf{w}}_i\right)$ are the eigenpairs of the preconditioned stabilized representer matrix $\left(\mathbf{R}^{-1/2}\mathbf{GDG}^{\mathrm{T}}\mathbf{R}^{-1/2}+\mathbf{I}\right)$. The array modes corresponding to the largest eigenvalues represent the interpolation patterns for the observations that are most stable with respect to changes in the innovation vector **d**, since the array modes depend *only* on the observation locations and not on the observation values.

As we discussed, over-fitting of the model to the observations can introduce non-physical features into the analysis. Therefore, it can be advantageous to exclude from the analysis array modes that are characterized by unphysical noise (see Moore et el., 2017). Practically, this can be achieving by appealing to the reduced-rank array modes and terminating the inner-loop calculations when the eigenvalues of the preconditioned stabilized representer matrix fall below a specific threshold. A useful rule-of-thumb is the "1% rule" of Bennett and McIntosh (1984).

The eigen-spectrum of the preconditioned stabilized representer matrix is computed on the fly during RBL4D-Var and can be plotted by first copying the NetCDF file **RBL4DVAR/EX3/wc13_mod.nc** into the directory **RBL4DVAR**, and then run script **WC13/plotting/plot_array_mode_spectrum.m**.

(i) Plot the eigen-spectrum and estimate where the number of inner-loops should be terminated based on the "1% rule." The dashed line in the plot represents $0.01\lambda_1$.

(ii) Now run the array mode driver as described below to compute the structure of one of the reduced-rank array modes that would be excluded from the analysis using the number of the inner-loops that you identified in (i).

(iii) Finally, repeat Exercise 3 by reducing the number of inner loops based on (i) and compare the increments with your original Exercise 3 calculation (optional).

Running the array mode driver

To run this exercise, go first to the directory **WC13/ARRAY_MODES,** and follow the directions in the **Readme** file. A change that you need to make is in **s4dvar.in**, where you will select the array mode that you wish to calculate (you may only calculate one mode at a time). The choice of array mode is determined by the parameter **Nvct**. The array modes are referenced in reverse order, so choosing **Nvct**=**Ninner** corresponds to the array mode with the largest eigenvalue, **Nvct**=**Ninner-1** is the array mode with second largest eigenvalue, and so on. Therefore, if you ascertain that the array modes after inner-loop 15 should not be included, you would choose **Nvct**=**Ninner-15**. In addition, set the value of **Nimpact** to be the number of the outer-loop that you want to compute the array modes for. In this example **Nimpact**=**1**. Also, be sure to set **Ninner** in **roms_wc13.in** to value that you used in Exercise 3.

Create a new subdirectory **EX7**, and save the solution in it for analysis and plotting to avoid overwriting solutions when playing with different CPP options and rerunning and recompiling:

```
mkdir EX7
mv Build_roms rbl4dvar.in *.nc log EX7
cp -p romsM roms_wc13.in EX7
```

where log is the ROMS standard output specified.

Plotting your results

Use **WC13/plotting/plot_array_modes.m** to plot a selection of fields for you chosen array mode.

References

Bennett, A. F. and P.C. McIntosh, 1984: Open ocean modelling as an inverse problem: M2 tides in Bass Strait. Journal of Physical Oceanography, **14**, 601–614.

Moore, A.M., H.G. Arango and C.A. Edwards, 2017: Reduced-Rank Array Modes of the California Current Observing System. Journal of Geophysical Research - Oceans, 10.1002/2017JC013172.