Bathymetry smoothing in ROMS: A new approach

Mathieu Dutour Sikirić

Ivica Janeković

Milivoj Kuzmić

Institute Rudjer Bošković, Zagreb

PLAN

I Problem set up
- Sigma coordinates.
- Slope factors
- ROMS model and vertical parametrization

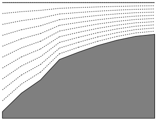
II Solution approaches
- Shapiro, Laplacian filters.
- Heuristic methods.
- Linear programming approaches.

III Comparison of selected methods
- Idealized cases
- Three grids of the Adriatic
- Nesting situation.

I. Problem set up

Sigma coordinate systems

- One way to deal with varying bathymetry: use σ-coordinates (Phillips 1957)



- On every cell e of bathymetry $h(e)$, choose a number N of vertical levels $h(e,k)$ for $1 \leq k \leq N$ with $h(e,0) = -h(e)$ and $h(e,N) = 0$.
- The differentiation rule of functions in σ-coordinate is

$$\left.\frac{\partial f}{\partial x}\right|_z = \left.\frac{\partial f}{\partial x}\right|_\sigma + \frac{\partial h}{\partial x}\frac{\partial f}{\partial \sigma}$$

- This creates a problem for horizontal derivatives, which become a difference of two terms. The wrong computation of the horizontal pressure gradient creates artificial currents.
- Smagorinsky 1967, Janjić 1977, Mesinger 1982, Haney 1991

The slope factors

- If e and e' are two adjacent wet cells, then

$$rx_0(h, e, e') = \frac{|h(e) - h(e')|}{h(e) + h(e')}$$

  The maximum over all such pairs is $rx_0(h)$, i.e. the Beckman & Haidvogel number.

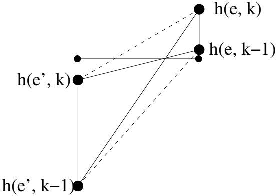
- If the vertical levels of the bathymetries are $h(e, k)$ for $1 \leq k \leq N$ then

$$rx_1(h, e, e', k) = \frac{|h(e, k) - h(e', k) + h(e, k-1) - h(e', k-1)|}{h(e, k) + h(e', k) - h(e, k-1) - h(e', k-1)}.$$

  The maximum over k and pairs e, e' of adjacent wet cells is $rx_1(h)$.
  This number is named hydrostatic inconsistency number or Haney number.

Hydrostatic consistency

- Denote by $C_k(e)$ the parallelepiped of water between depth $h(e, k-1)$ and depth $h(e, k)$.
- Hydrostatic consistency means that if e and e' are any two adjacent cells, then $C_k(e)$ and $C_k(e')$ share a level.



- To impose that $C_k(e)$ and $C_k(e')$ share a level is equivalent to $rx_1(h, e, e', k) \leq 1$ (Rousseau and Pham 1971, Mesinger 1982, Haney 1991).
- This requirement is very strong and almost impossible to fulfill.

What are the right values of rx_1, rx_0

There is no general agreement on this question

- The factor which matters for the horizontal pressure gradient is the Haney number $rx_1(h)$.
- It is extremely difficult to achieve $rx_1(h) \leq 1$.
- In Mellor-Ezer-Oey, 1994 it is argued that the HPG error is not very important and disappears after running the model for some time.
- Kliem-Pietrzak, 1999 contests this for the Skagerrak region.
- Sasha Shchepetkin, 2008 says that $rx_1(h) \leq 3$ is "safe", $rx_1(h) \simeq 5$ is "common" and $rx_1(h) \geq 8$ is "insane".
- Kate Hedström, 2008 reported no problem with $rx_1(h) \simeq 16$.
- We experienced blow ups with grids with $rx_1(h) \geq 9$.
- We call a grid numerically stable if $rx_1(h) \leq 6$.

The ROMS model

- ROMS is an hydrostatic regional σ-coordinate ocean model with several advection scheme.
- There are three versions of ROMS
  - ROMS AGRIF maintained by Debreu (public).
  - ROMS UCLA maintained by Shchepetkin (non public).
  - ROMS Rutgers maintained by Arango (public, main version).
- ROMS Rutgers has possibility of coupling with the SWAN model, adjoint and tangent linear functionalities for strong 4dvar, weak 4dvar.
- ROMS AGRIF and ROMS UCLA have built in nesting capabilities. ROMS UCLA has the highest speed and there exists a non hydrostatic version of ROMS UCLA.

Vertical parametrization in ROMS

- The ROMS vertical parametrization depends on three parameters hc, θ_s, θ_b

$$h(e,k) = s_w(k)hc + (h(e) - hc)c_w(k).$$

hc is the thermocline parameter and it is lower than the minimal depth of the model.

- The vertical parametrization function depends on θ_s and θ_b and is $s_w(k) = -\frac{k}{N}$.

$$c_w(k) = (1-\theta_b)\frac{\sinh \theta_s s_w(k)}{\sinh \theta_s} + \theta_b \left\{ \frac{\tanh \theta_s (s_w(k) + \frac{1}{2})}{2 \tanh \frac{\theta_s}{2}} - \frac{1}{2} \right\}$$

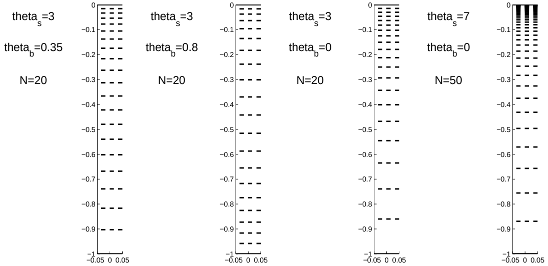
This formula is relatively arbitrary (Song, 1994) and another one may work just as well.

- If $hc = 0$ then we have $h(e,k) = h(e)c_w(k)$ and we get

$$rx_1(h) = \max_{1 \leq k \leq N} \frac{c_w(k) + c_w(k-1)}{c_w(k) - c_w(k-1)} rx_0(h)$$

Choice of vertical stratification

- If one wants only to minimize rx_1, then the choice is $\theta_b = 0$ and θ_s high, i.e. concentrate the vertical levels on the surface. But we cannot concentrate the levels too much on the surface so it is recommended to have θ_s at most 7.
- If the bottom boundary layers is a zone of interest, then a nonzero value of θ_b has to be chosen.
- The number of vertical levels is the main constraints. It limits the computational possibilities and the error pressure gradient.



Possible ways to deal with the problem

If the bathymetry is too steep then this causes instabilities and inaccuracies. Some possible ways to deal with it:

- Use a high order pressure gradient scheme (Chu & Fan, 1997, 1998, 2003) (computational price)
- Adjust the vertical stratification, i.e. s_w, c_w and in case of ROMS θ_s, θ_b (modelling choices).
- Decrease the number N of vertical levels (less realistic)
- Make the horizontal grid finer (computational price).
- Smooth the bathymetry (less realistic).
- Use a z- or generalized coordinate system (change of model).

We consider the smoothing methods to reduce the magnitude of the problem.

II. Solution approaches

The goal

- The grid is build in the following way:
  - Build an initial grid using coastline informations.
  - Choose the parameters N, θ_s, θ_b and hc.
  - Find the initial bathymetry h^{obs} from existing data set (NOAA, Gshhs, Gebco, etc.)
  - Determine the smoothed bathymetry h.
- Requirements:
  - $rx_0(h)$ and $rx_1(h)$ low.
  - The "distance" between h and h^{obs} small.
  - h should have the same physical characteristics as h^{obs}.
- For a given r and h^{obs}, we will present methods to get h with $rx_0(h) \leq r$.
- The analysis for rx_1 works similarly.

Why optimize with rx_0?

- It is less expensive computationally than optimizing with rx_1.
- If the depth is sufficiently large then the relation

$$rx_1(h) = \max_{1 \le k \le N} \frac{c_w(k) + c_w(k-1)}{c_w(k) - c_w(k-1)} rx_0(h)$$

  is almost exact.
- For the interesting domain $rx_1 \le 6$, the result of optimization with rx_1 is at most 5% different from optimizing with respect to rx_0.
- The error pressure gradient is more important in region of high density differences. Those are typically regions of moderate to high depth.

The grid problem

- The observed bathymetry is available on a grid different from the one of the model.
- We have two possible situations:
  - The observations are sparser than the model bathymetry.
  - The resolution of the observations is better than the resolution of the model.
- In the first situation, the solution is necessarily to interpolate the bathymetry from the available observations to the model grid. If the observations have no regular structure (ship, ...) then the best is to use natural neighbor interpolation (Sibson 1981) with the program `nnbathy` by Pavel Sakov, 2006.
- In the Adriatic, actually we have high resolution bathymetry and we are in the latter situation.

The averaging procedure

- When we have more bathymetry observations than necessary for the model, then we take the average over wet cells of the observations that belong to them.
- Another strategy is to use the Shapiro filter several times and then to interpolate to the grid of the model.
- The problem is that by doing this we smooth the bathymetry and reduce the slope factor even when this is not needed.
- The HPG problem is a model problem and so it should be treated at level of grid of the model.
- The right method is thus:
  - First compute of the model at the grid level.
  - Then smooth it, to reduce the HPG error.

The Shapiro filter

- It is a filter designed to smooth out fast waves in finite difference models (Shapiro 1975).
- It was not designed for smoothing out the bathymetry but it is still frequently used to smooth out bathymetry variations.
- Every ROMS version has its own version of the filter.
  - ROMS AGRIF has a Shapiro filter applied to the logarithm of the bathymetry first in x-direction and then in y-directions.
  - ROMS UCLA has a Shapiro filter applied to the logarithm of the bathymetry with a more complex stencil.
  - ROMS Rutgers has a Shapiro filter applied in x- and y-directions only to the points where the slope factor is not correct.
- The POM model has its own filter named "Gaussian filter".

The Shapiro filter of ROMS Rutgers

- It is applied to the bathymetry in the following way:

  $h \leftarrow h^{obs}$

  while $rx_0(h) > r$ **do**

      $h' \leftarrow$ Shapiro filtering of h on x direction.

      for e in wet cells **do**

          if $rx_o(h, e) > r$ **then**

              $h(e) \leftarrow h'(e)$

          end if

      end for

      Do the same in y direction

  end do

- For some bathymetries the Shapiro filter converges to h with $rx_0(h) > r$ and thus the program never ends.
- The best Shapiro filter of all 3 is the one of ROMS UCLA.

Laplacian filter

- It works in the following way:
  - start with $h = h^{obs}$.
  - If $rx_0(h, e) > r$ we do:

$$h(e) \leftarrow h(e) + \frac{1}{2N(e)} \sum_{e' \in N(e)} \{h(e') - h(e)\}$$

    with $N(e)$ the set of wet cells adjacent to the wet cell e.
  - Iterate until $rx_0(h) \leq r$.
- This filter is more stable than Shapiro filter, but there is a still a problem of having the program end.
- Shapiro filter and Laplacian filter are very frequently used but they are not very good methods.

The Martinho & Batteen (MB) scheme

- Whenever the slope is not correct the chosen solution (Martinho & Batteen 2006) is to increase the bathymetry.
  - Start with $h = h^{obs}$
  - If
$$\frac{h(e) - h(e')}{h(e) + h(e')} > r \quad \text{then} \quad h(e') \leftarrow \frac{1-r}{1+r} h(e)$$
  - All pairs (e, e') are considered iteratively until the slope factor is correct. The result is independent of the order of operations.
- They also proposed to preserve the volume by replacing the bathymetry h obtained by their method by
$$h \leftarrow h \frac{\text{vol } h^{obs}}{\text{vol } h}.$$

This method works because $rx_0(\alpha h) = rx_0(h)$.

The bathymetry decreasing scheme

- Whenever the slope is not correct the chosen solution is to decrease the bathymetry.
  - Start with $h = h^{obs}$
  - If
$$\frac{h(e) - h(e')}{h(e) + h(e')} > r \quad \text{then} \quad h(e) \leftarrow \frac{1+r}{1-r} h(e')$$
  - All pairs (e, e') are considered iteratively until the slope factor is correct. The result is independent of the order of operations.
- This filter can help for initialization problems when the initial state has to be taken from measured or initial values and not from extrapolated values from a nearest point.

The Mellor-Ezer-Oey (MEO) scheme

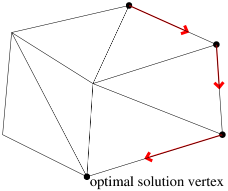
- (Mellor 1994) If we want to preserve volume, then another scheme is possible.
  - If we have
    $$\frac{h(e) - h(e')}{h(e) + h(e')} > r$$
    then we write
    $$h(e) \leftarrow h(e) - \frac{V(e, e')}{A(e)} \quad \text{and} \quad h(e') \leftarrow h(e') + \frac{V(e, e')}{A(e')}$$
    with $V(e, e')$ adjusted so that $\frac{h(e)-h(e')}{h(e)+h(e')} = r$ and $A(e)$, $A(e')$ the area of wet cell e, e'.
  - All pairs (e, e') of adjacent wet cells are considered iteratively until the bathymetry is correct.
- A priori, the final bathymetry depends from the order of the operations.

Definition of linear programs

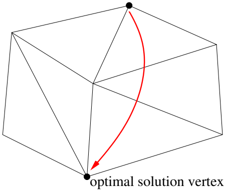
- A linear program is the problem of maximizing a linear function $f(x)$ over a set $\mathcal{P}$ defined by linear inequalities.

$$\mathcal{P} = \{x \in \mathbb{R}^d \quad \text{such that} \quad f_i(x) \geq b_i\}$$

  with f_i linear and $b_i \in \mathbb{R}$.
- The solution of linear programs is attained at vertices of $\mathcal{P}$.
- There are two classes of solution methods:



Simplex method



Interior point method

Linear programming methods

- The inequality $rx_0(h, e, e') \leq r$ corresponds to:

$$-r(h(e) + h(e')) \leq h(e) - h(e') \leq r(h(e) + h(e'))$$

- We introduce some auxiliary variable $\delta(e)$ with

$$|h(e) - h^{obs}(e)| \leq \delta(e) \quad \text{i.e.} \quad \pm\left(h(e) - h^{obs}(e)\right) \leq \delta(e)$$

- And we minimize

$$\sum_e \delta(e) \quad \text{that is} \quad \sum_e |h(e) - h^{obs}(e)|.$$

- There are many possible variants, which are still in the linear programming paradigm:
  - Preserve the total volume of the basin.
  - Have a different objective function.
  - Impose only positive/negative corrections at some points.
  - Impose maximum amplitude condition.
  - Fix some points (nested applications)

Linearized Mellor-Ezer-Oey (LMEO) scheme

- For all pair (e, e') of adjacent wet cells e, e' consider the operations.

$$h(e) \leftarrow h(e) - \frac{V(e, e')}{A(e)} \quad \text{and} \quad h(e') \leftarrow h(e') + \frac{V(e, e')}{A(e')}$$

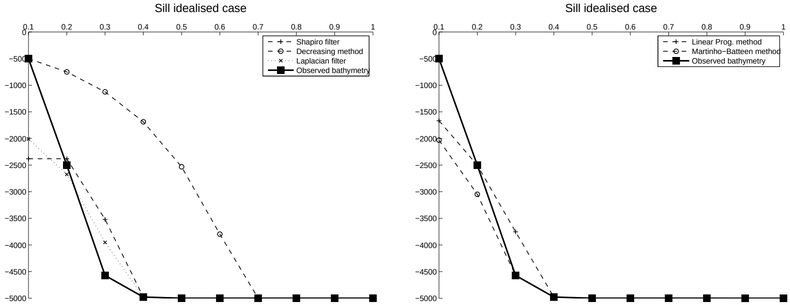
- For a set of volumes $V(e, e')$, consider the resulting bathymetry h obtained from h^{ave}.
- The objective function is

$$\sum_{(e,e')} |V(e, e')|$$

- This method is supposed to be a reformulation of Mellor-Ezer-Oey where we minimize the volumes $V(e, e')$ involved.

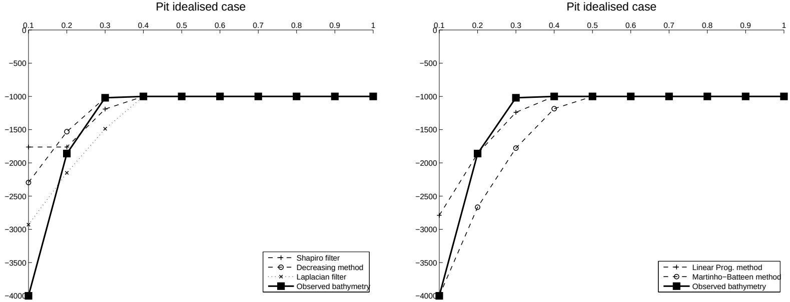
III. Comparison of selected methods

Idealized cases: Sill



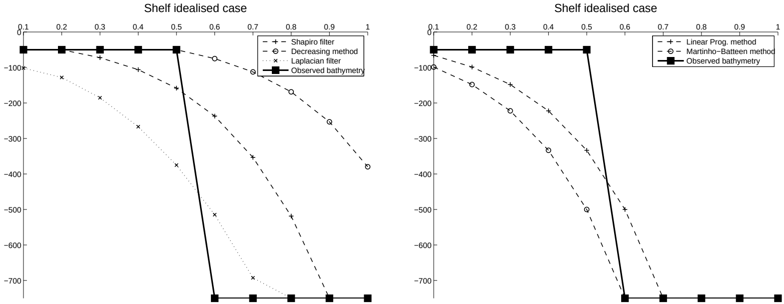
- We should avoid the bathymetry decreasing method.

Idealized cases: Pit



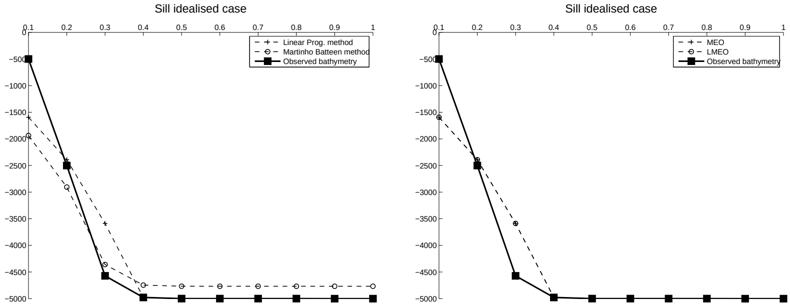
- The Martinho & Batteen method is good for preserving the depth of the pits.

Idealized cases: Shelf break



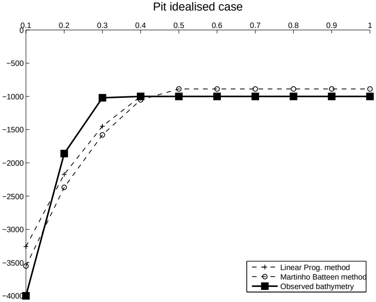
- LP does a better job of preserving the shelf break.

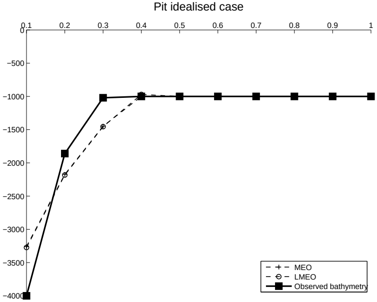
Idealized cases (volume preserving): Sill



- MEO (Mellor-Ezer-Oey) gives the same result as LMEO (Linearized Mellor-Ezer-Oey).
- MB method spread the perturbation globally.

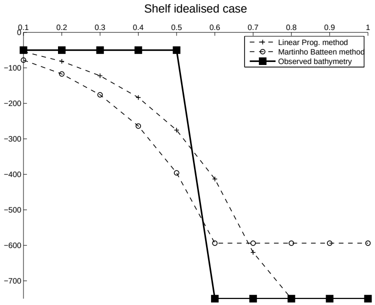
Idealized cases (volume preserving): Pit

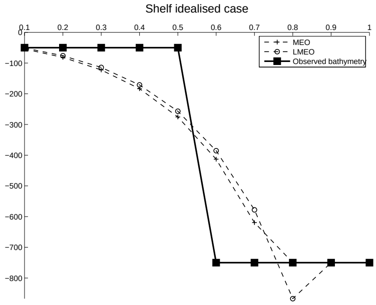




- The MB method is again better for the pits but the perturbation is global to the basin.

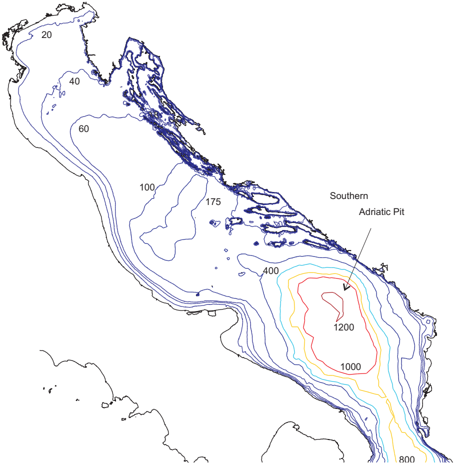
Idealized cases (volume preserving): Shelf break





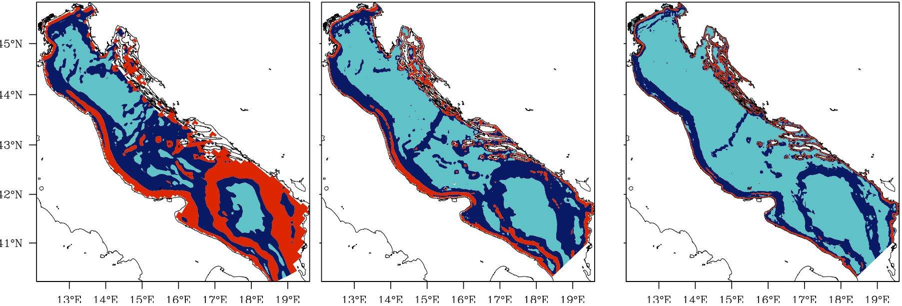
- The Linearized Mellor-Ezer-Oey method gives a worse result than the MEO method. We should avoid LMEO.
- LP is very near to the MEO method.

The Adriatic Sea



- The bathymetry is highly varying and the coastline is diverse.
- We chose three grids 160×60, 127×368, 271×751

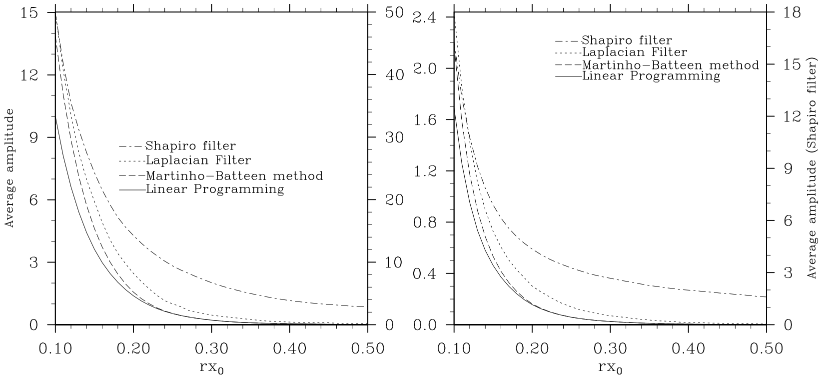
Hydrostatic consistency & numerical stability



60×160 127×368 271×751

The regions of hydrostatic consistency & numerical stability ($rx_1(h,e) \leq 1$ in light blue), hydrostatic inconsistency & numerical stability ($1 \leq rx_1(h,e) \leq 5$ in dark blue) and hydrostatic inconsistency & numerical instability ($rx_1(h,e) \geq 5$ in red)

Average amplitude of bathymetry modification

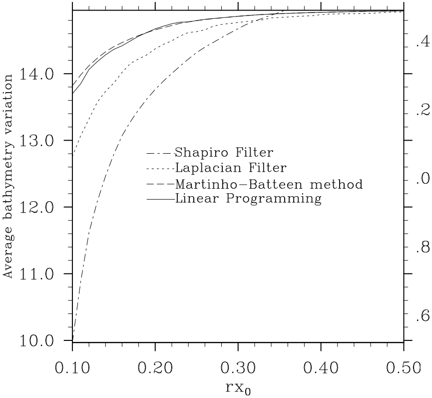


60 × 160

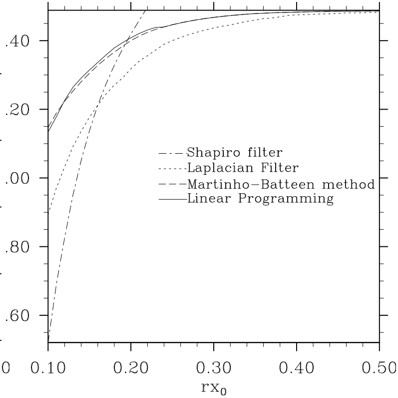
127 × 368

The average amplitude of bathymetry modification (m) in terms for bathymetry smoothing methods

Average variation of bathymetry



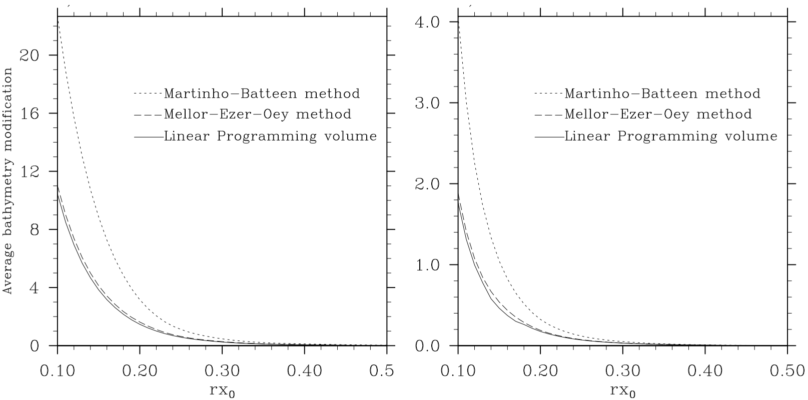
60 × 160



127 × 368

The average variation of the bathymetry (m) from wet cell to wet cell for bathymetry smoothing methods in terms of $rx_0(h)$

Average amplitude of bathymetry modification

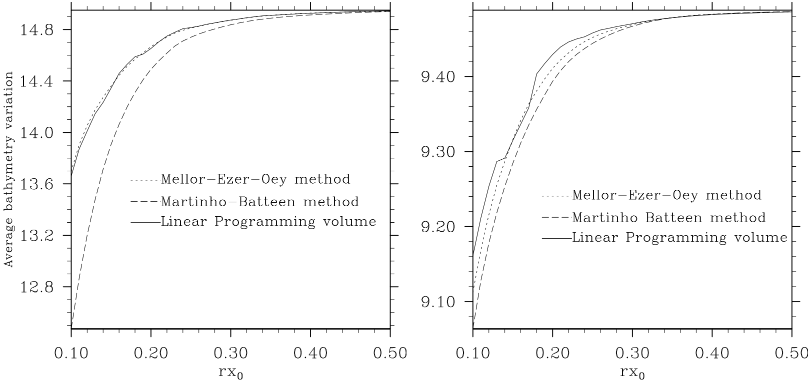


60 × 160

127 × 368

The average amplitude of bathymetry modification (m) in term of rx_0 for volume preserving smoothing methods

Average variation of the bathymetry



60×160 $\qquad$ 127×368

The average variation of the bathymetry (m) from wet cell to wet cell for bathymetry smoothing methods preserving volume in terms of rx_0

Effect of smoothing

- The need for smoothing decrease when the horizontal grid is finer:

grid	volume perturbed
60×160	322km^3
127×368	20km^3
271×751	7.2km^3

- Time runs:
  - Heuristic methods take at most 20 seconds for smoothing.
  - Shapiro and Laplacian do not take more than a few minutes in general.
  - Linear programming takes more time 5 min for 60×160, 1 hour for 127×368 and 1 day for 271×751.
- Having the right bathymetry in the model can be the key to correct modelization:
  - ➠ Batteen et al., 2007. *A process oriented modelling study of the coastal Canary and Iberian Current system*. Ocean modelling 18, 1–36.

Stability of solutions

What happens if one perturb by an infinitesimal quantity the observed bathymetry and/or the slope factor?

- Heuristic methods (MEO, MB) are continuous.
- Shapiro filter and Laplacian filter methods are not continuous.
- Linear programming methods are not continuous since there are possible hoppings from one vertex to an adjacent one.

In practice during 0.01 increments to rx_0 for the 127×368 grid,

method	average change	maximal change
MB	5.4cm	7.8m
LP	5.2cm	13.8m
Laplacian	6.4cm	23m
Shapiro	$40cm$	28m

Nested grid situations

- Suppose we have a grid of say $2km$ of resolution, another grid of $700m$ around Rovinj region is embedded in it in a 1-way coupling situation. We want the bathymetry of the embedded grid to coincide with the bathymetry of the embedded grid on the boundaries.
- The method is the following.
  1. Compute the raw bathymetry h^{raw} of the embedded grid using averaging operations.
  2. Interpolate the bathymetry of the $2km$ grid to the $700m$ grid.
  3. Fix the boundary values of the bathymetry of the $700m$ grid to be interpolated values.
  4. Smooth h^{raw} by specifying no change of boundary values.

Conclusions

- Shapiro and Laplacian filter should be avoided since they create large perturbation of the bathymetry.
- Heuristic methods like Martinho-Batteen, Mellor-Ezer-Oey work relatively well.
- If $rx_0(h) \leq 0.2$ is needed, then linear programming might be what you need.
- All programs (in matlab) for optimizing over rx_0 or rx_1 are available from

  `http://drobilica.irb.hr/~mathieu/Bathymetry/index.html`