ALE moving mesh tsunami simulation. Application to the case study of the Tohoku tsunami

L.Arpaia, M.Ricchiuto

Inria Bordeaux Sud-Ouest, Team CARDAMOM

SIAM GS 2017, September 14, 2017 • Erlangen, Germany

INVENTEURS DU MONDE NUMÉRIQUE

Increasing the accuracy of tsunami simulations

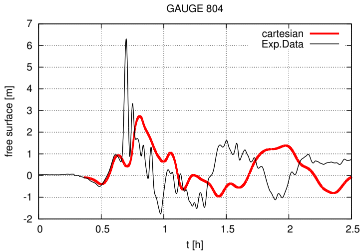
- For large scale tsunami simulations it is very hard to capture all flow scales in a single computation
- Static grids are inherently not efficient. To enhance the resolution of wave patterns during propagation and shoaling or runup stages in which fine scale flooding occurs ⇒ **mesh adaptation**
- *h*-refinement is the most widely used technique to refine mesh size. Cartesian grids: AMR *[Berger and Colella, 1989]*. Unstructured grids: local remeshing *[Alauzet et al., 2007; Isola and Guardone, 2011]*
- AMR already implemented in a few geophysical codes: `GeoClaw` *[Berger et al., 2011]*, `Gerris Flow` *[Popinet, 2011]*

Moving mesh methods

- Although very powerful, remeshing techniques require much higher overheads and a complex data structure
- Focus on **r-adaptation** techniques. Nodes number is fixed and redistributed in the domain.
- Data structure does not change and nodal movement is obtained by solving efficiently an elliptic PDE *[Huang and Russell, 1994]*
- the potential shown to capture shocks, boundary layers and singularities *[Ceniceros and Hou, 2001]*
- efficient conservative/accurate remaps of flow variables: **Arbitrary Lagrangian Eulerian** (ALE) coupling *[Donea, 1989; Cao et al., 1999]* and rezoning *[Tang and Tang, 2003]*

Earth curvature effects

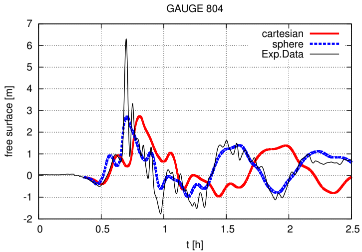
- Earth curvature affects large scale tsunami dynamics
- we set the **SWEs on a rotating sphere** (standard model for General Circulation Model)
  - SWEs in curvilinear coordinates
- extension of the ALE moving mesh method in curvilinear coordinates *[Auber et al., 2008]*



Example: numerical simulation against recorded data at a gauge station G804 for the Tohoku tsunami

Earth curvature effects

- Earth curvature affects large scale tsunami dynamics
- we set the **SWEs on a rotating sphere** (standard model for General Circulation Model)
  - SWEs in curvilinear coordinates
- extension of the ALE moving mesh method in curvilinear coordinates *[Auber et al., 2008]*

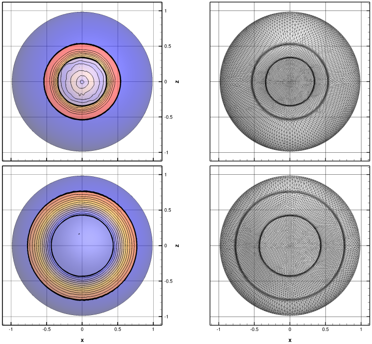


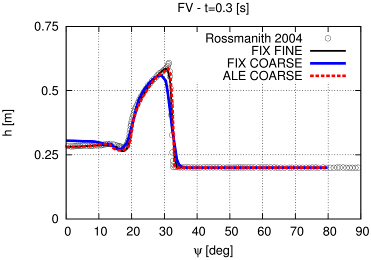
Example: numerical simulation against recorded data at a gauge station G804 for the Tohoku tsunami

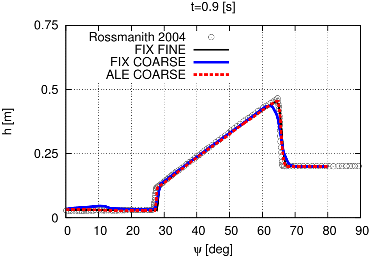
Illustrative example: circular hump on a sphere

RD	# POINTS	TIME[s]
FIX-COARSE	7122	34.58
FIX-FINE	39699	485.37
ADAPT-ALE	7122	201.05
Rossmanith	34680	-

$$\% \text{ MMPDE of tot CPU } = 37\%$$

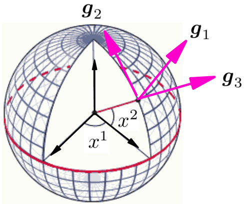






SWEs in curvilinear coordinates

Consider **covariant basis vectors** $\mathbf{g}_1, \mathbf{g}_2$ with $G_{ij} = \mathbf{g}_i \cdot \mathbf{g}_j$ and $\sqrt{G} = \sqrt{\det(\mathbf{G})}$. SWEs in covariant Eulerian framework:



$$\frac{\partial \mathrm{u}}{\partial t} + \frac{1}{\sqrt{G}} \frac{\partial \mathrm{F}^j}{\partial x^j} = \mathrm{S}(\mathbf{x}, \mathrm{u}), \quad \mathrm{u} = \begin{bmatrix} h \\ hu^i \end{bmatrix},$$

$$\mathrm{F}^j = \sqrt{G} \begin{bmatrix} hu^j \\ T^{ij} \end{bmatrix}, \quad \mathrm{S} = - \begin{bmatrix} 0 \\ G^{ij} gh \frac{\partial b}{\partial x^i} \end{bmatrix} - \begin{bmatrix} 0 \\ \Gamma^i_{jk} T^{jk} \end{bmatrix}$$

with $T^{ij} = hu^i u^j + \frac{1}{2} gh^2 G^{ij}$. It admits the lake at rest

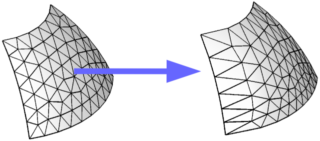
$$\frac{1}{\sqrt{G}} \frac{\partial \mathrm{F}^j}{\partial x^j} = \mathrm{S} \quad \Rightarrow \quad \eta = h + b = const.$$

Property. We call **Well Balanced** a scheme which is able of preserving exactly the "lake at rest" state. *[Ullrich et al., 2010]*

ALE-SWEs in curvilinear coordinates

Given the mapping $A(t) : \mathcal{A}_0 \to \mathcal{A}(t)$ with $\mathbf{x} = A(\chi, t)$ with

$$\mathcal{J}_A = \frac{\partial \mathbf{x}}{\partial \chi}, \; J_A = \det \mathcal{J}_A, \quad \left.\frac{\partial x^j(t)}{\partial t}\right|_{\chi} = \sigma^j(\mathbf{x}, t)$$



SWEs in ALE framework and curvilinear coords writes

$$\left.\frac{\partial}{\partial t}\right|_{\chi} \left(\sqrt{G} J_A \mathrm{u}\right) + J_A \frac{\partial}{\partial x^j} \left(\mathrm{F}^j - \sqrt{G} \sigma^j \mathrm{u}\right) = \sqrt{G} J_A \mathrm{S}$$

From continuum mechanics $\Rightarrow$ Geometric Conserv. Law (GCL)

$$\left.\frac{\partial \sqrt{G} J_A}{\partial t}\right|_{\chi} = J_A \frac{\partial \sqrt{G} \sigma^j}{\partial x^j} \quad \Rightarrow \quad \left.\frac{\partial}{\partial t}\right|_{\chi} \int_V \sqrt{G} d\mathbf{x} = \int_{\partial V} \boldsymbol{\sigma} \cdot \mathbf{n} \sqrt{G} ds$$

Property. We call **DGCL** (Discrete GCL) a scheme which is able of verifying exactly the GCL. *[Mavripils and Yang, 2002]*

Well Balanced ALE form

Given the **ALE remap** equation for the function $b(\mathbf{x}(t))$

$$\left.\frac{\partial b(\mathbf{x}(t))}{\partial t}\right|_{\chi} = \frac{\partial b}{\partial x^j}\left.\frac{\partial x^j(t)}{\partial t}\right|_{\chi} \Rightarrow \left.\frac{\partial(\sqrt{G}J_A b)}{\partial t}\right|_{\chi} = J_A\frac{\partial}{\partial x^j}\left(\sqrt{G}b\sigma^j\right)$$

the balance law in ALE framework in the new variable $\mathrm{u}_\eta = [\eta\ hu^i]$

$$\left.\frac{\partial}{\partial t}\right|_{\chi}\left(\sqrt{G}J_A\mathrm{u}_\eta\right) + J_A\frac{\partial}{\partial x^j}\left(\mathrm{F}^j - \sqrt{G}\sigma^j\mathrm{u}_\eta\right) = \sqrt{G}J_A\mathrm{S}$$

In case of constant state η_0, we have:

$$\mathrm{u}_\eta\underbrace{\left(\left.\frac{\partial\sqrt{G}J_A}{\partial t}\right|_{\chi} - J_A\frac{\partial\sqrt{G}\sigma^j}{\partial x^j}\right)}_{\text{DGCL}} + J_A\underbrace{\left(\frac{\partial \mathrm{F}^j}{\partial x^j} - \sqrt{G}\mathrm{S}\right)}_{\text{Well Balanced}}$$

$$+\sqrt{G}J_A\underbrace{\left(\left.\frac{\partial \mathrm{u}_\eta}{\partial t}\right|_{\chi} - \sigma^j\frac{\partial \mathrm{u}_\eta}{\partial x^j}\right)}_{=0\leftarrow\eta=const} = 0$$

Mass conservation issue

Solving numerically the ALE remap for b

$$\left.\frac{\partial_t(J_A b)}{\partial t}\right|_\chi = J_A \nabla \cdot (b\boldsymbol{\sigma}) \quad \Rightarrow \quad \left.\frac{\partial}{\partial t}\right|_\chi \int_\Omega b d\mathbf{x} = \int_{\partial\Omega} b\boldsymbol{\sigma} \cdot \mathbf{n} = 0$$

introduce **large numerical errors** in the computation of nodal values b_i [Zhou et al., 2013]

Mass conservation issue

Recompute at each time step the nodal values by $b(\mathbf{x}(t)) = b(A(\mathbf{X}, t))$

$$\left.\frac{\partial_t (J_A b)}{\partial t}\right|_{\chi} \neq J_A \nabla \cdot (b\boldsymbol{\sigma}) \quad \Rightarrow \quad \left.\frac{\partial}{\partial t}\right|_{\chi} \int_{\Omega} b d\mathbf{x} \neq \int_{\partial\Omega} b\boldsymbol{\sigma} \cdot \mathbf{n} = 0$$

Remark. This choice computes accurate values of $b(\mathbf{x}(t))$ but does not conserve mass:

$$\left.\frac{\partial}{\partial t}\right|_{\chi} \int_{\Omega} \eta d\mathbf{x} + \int_{\partial\Omega} h\mathbf{u} \cdot \mathbf{n} = 0$$

We can drive this error to zero increasing the order of quadrature formula:

$$\int_{\Omega} b d\mathbf{x} \approx \sum_{K \in \mathcal{T}_h} \int_K b_h \, d\mathbf{x} = \sum_{K \in \mathcal{T}_h} |K^{n+1}| \sum_{q=1}^{N_q} \omega_q b_q$$

$$\mathbf{b}_i^{n+1} = \frac{1}{|C_i^{n+1}|} \sum_{K \in \mathcal{D}_i} |K^{n+1}| \sum_{q=1}^{N_q} \omega_q b_q$$

Mass conservation issue

Recompute at each time step the nodal values by $b(\mathbf{x}(t)) = b(A(\mathbf{X}, t))$

$$\left.\frac{\partial_t (J_A b)}{\partial t}\right|_{\chi} \neq J_A \nabla \cdot (b\boldsymbol{\sigma}) \quad \Rightarrow \quad \left.\frac{\partial}{\partial t}\right|_{\chi} \int_{\Omega} b d\mathbf{x} \neq \int_{\partial\Omega} b\boldsymbol{\sigma} \cdot \mathbf{n} = 0$$

Remark. This choice computes accurate values of $b(\mathbf{x}(t))$ but does not conserve mass:

$$\left.\frac{\partial}{\partial t}\right|_{\chi} \int_{\Omega} h d\mathbf{x} + \int_{\partial\Omega} h\mathbf{u} \cdot \mathbf{n} = -\left.\frac{\partial}{\partial t}\right|_{\chi} \int_{\Omega} b d\mathbf{x} \neq 0$$

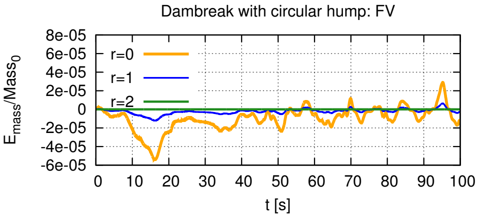
We can drive this error to zero increasing the order of quadrature formula:

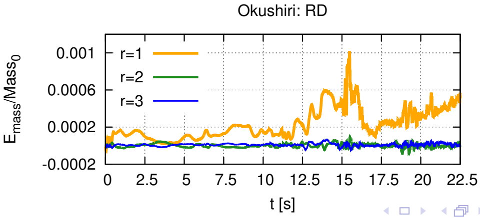
$$\int_{\Omega} b d\mathbf{x} \approx \sum_{K \in \mathcal{T}_h} \int_K b_h \, d\mathbf{x} = \sum_{K \in \mathcal{T}_h} |K^{n+1}| \sum_{q=1}^{N_q} \omega_q b_q$$

$$\mathbf{b}_i^{n+1} = \frac{1}{|C_i^{n+1}|} \sum_{K \in \mathcal{D}_i} |K^{n+1}| \sum_{q=1}^{N_q} \omega_q b_q$$

Mass conservation issue

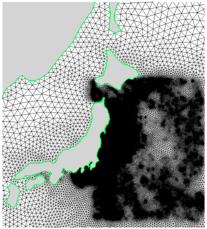
$$E_{mass} = \int_{\Omega} h d\mathbf{x} - \int_{\Omega_\chi} h_0 d\boldsymbol{\chi} + \int_{\tau=0}^{t} \int_{\partial\Omega} h\mathbf{u} \cdot \mathbf{n} \, ds \, d\tau$$

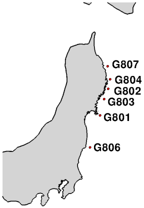


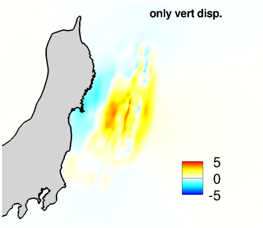


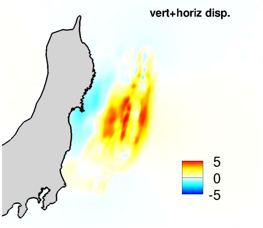
Tohoku-Honsu tsunami

- tsunami source: *[Satake, 2012]* with/without horizontal displacement
- bathymetry: resolution 120 *m*
- numerical solution recorded at 6 nearshore GPS gauges

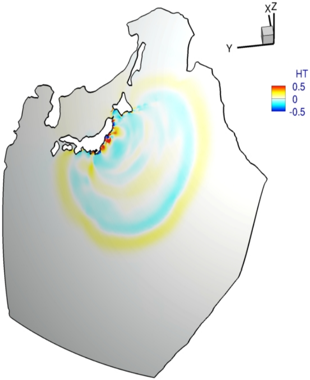


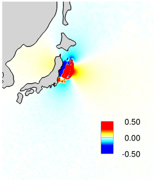




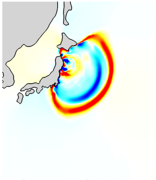


Tohoku-Honsu tsunami

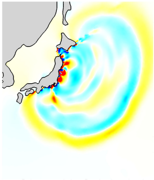




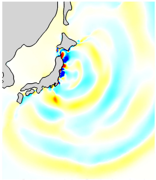
14:51:18 JST



+1h



+2h

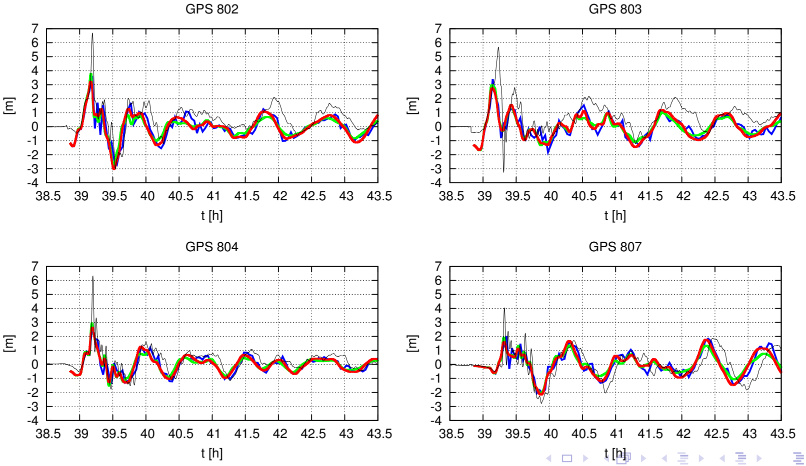


+3h

Fixed mesh: comparison with other codes

Mesh: 1,080,181 triangles. $h_K = 120 \div 50000\ m$
Source: only vertical displacement

— RD — FV — Telemac-2D — Obs.Data

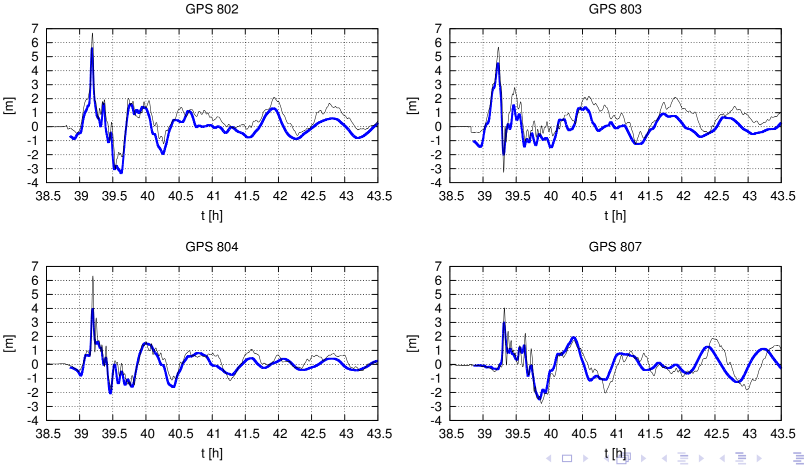


Fixed mesh: reference solution

Mesh: 1,379,054 triangles. $h_K = 120 \div 5000\ m$

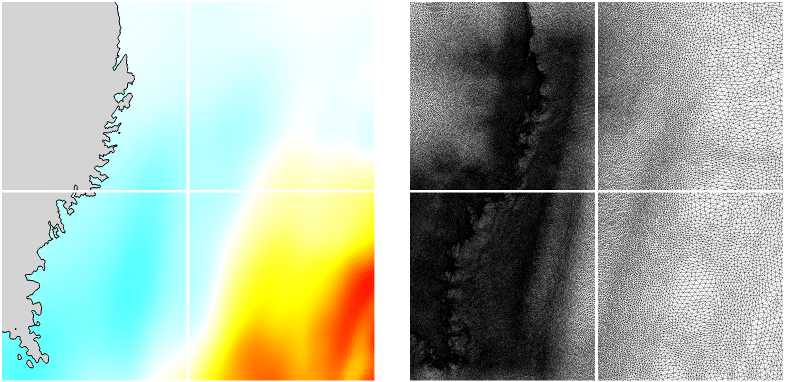
Source: horizontal+vertical displacement

— RD — Obs.Data



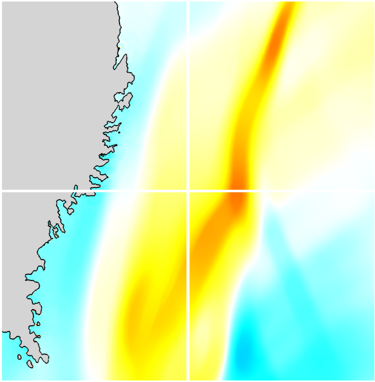
ALE simulation, $t = 14:51:18$ JST

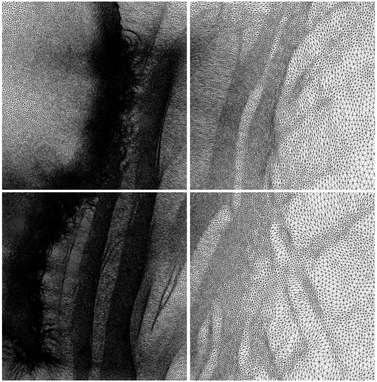
Mesh: 728,874 triangles. $h_K = 360 \div 15000\ m$
Source: horizontal+vertical displacement



ALE simulation, $t = +10'$

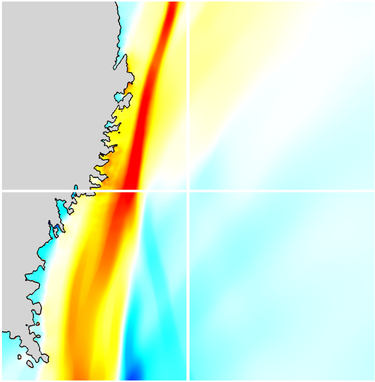
Mesh: 728,874 triangles. $h_K = 360 \div 15000\ m$
Source: horizontal+vertical displacement

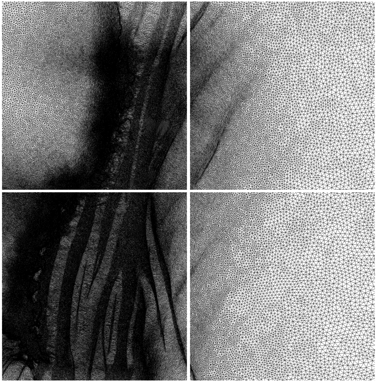




ALE simulation, $t = +20'$

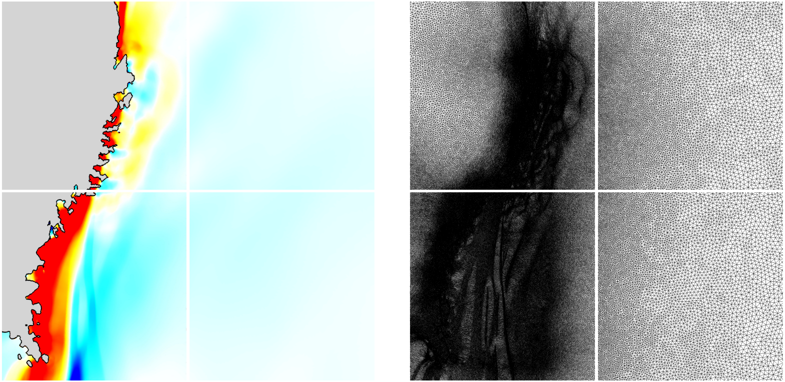
Mesh: 728,874 triangles. $h_K = 360 \div 15000\ m$
Source: horizontal+vertical displacement





ALE simulation, $t = +30'$

Mesh: 728,874 triangles. $h_K = 360 \div 15000\ m$
Source: horizontal+vertical displacement

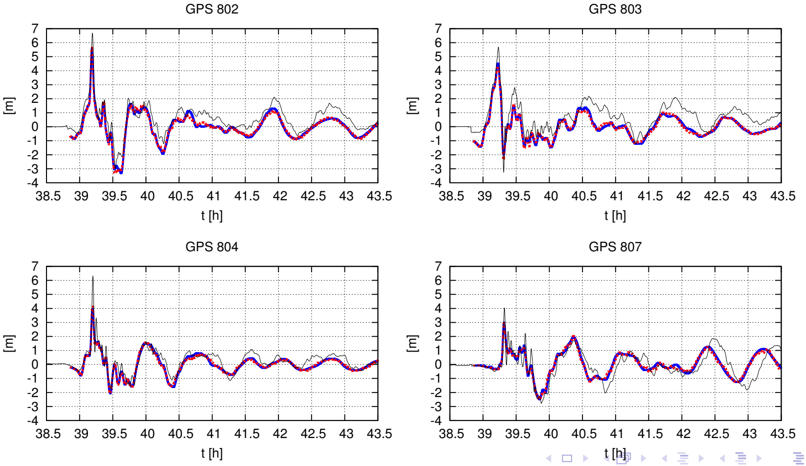


ALE simulation against reference

Mesh: 728,874 triangles. $h_K = 360 \div 15000\ m$

Source: horizontal+vertical displacement

— Reference — ADAPT-ALE — Obs.Data



Limitations and future perspectives

ALE-moving mesh method can be effective for tsunami simulations.

- the MMPDE of *[Ceniceros and Hou, 2001]* is simple but poses **strong limitations on the control on mesh quality**, specially when refining along complex flow feature such as flooding on irregular coastlines or complex smooth wave patterns
  $\Rightarrow$ other MMPDE can be tested/used:
  - matrix monitor function *[Huang, 2006]*,
  - elastic mesh model *[Stein et al., 2004]*
  - Monge-Ampere MMPDE *[Budd and Williams, 2009]*
- numerical treatment of wetting/drying in presence of moving mesh
- sensitivity to the moving mesh parameters