Well balanced ALE : on time dependent adaptation for shallow water flows

L. Arpaia and M. Ricchiuto

INRIA Bordeaux - Sud-Ouest, project BACCHUS

SIAM conference on
Nonlinear Waves and Coherent Structures

August 11th - 14th, 2014

Churchill College, University of Cambridge

Scope and outline of the talk

Main objective

Discuss techniques allowing time dependent mesh adaptation for shallow water flows in a cost effective manner :

1. simple to implement, no major code restructuring, no major modifications to the scheme
2. minimize error *vs* CPU time

Milestones

- Simple mesh deformation based on solution smoothness
- ALE formulation for balance laws *vs* steady invariants
- Scheme-mesh adaptation coupling based on two strategies
  1. Deformation-Projection-Evolution (DPE)
  2. Deformation-ALE evolution (DALE)

Outline

Mathematical setting

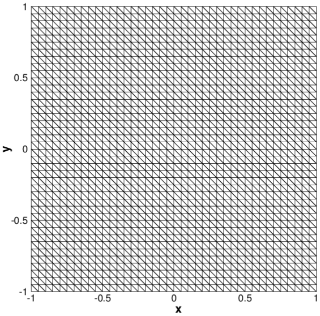
Model equation

Seek approximate solutions of

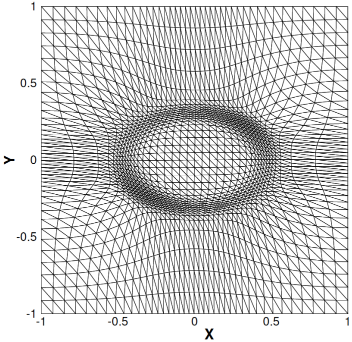
$$\partial_t u + \nabla \cdot \boldsymbol{\mathcal{F}}(u) = \mathcal{S}(u,\, g(\vec{x})) \qquad (1)$$

on a time dependent unstructured mesh $\mathcal{T}_h(t)$.

$\mathcal{T}_h(t = t_1)$



$\mathcal{T}_h(t = t_2)$



Mathematical setting

Model equations

Seek approximate solutions of

$$\partial_t u + \nabla \cdot \boldsymbol{\mathcal{F}}(u) = \mathcal{S}(u,\, g(\vec{x})) \tag{1}$$

on a time dependent unstructured mesh $\mathcal{T}_h(t)$.

Remarks

- We focus on triangular meshes
- Equation (1) assumed to admit non-trivial steady equilibria characterized by

$$\eta(u) = \eta_0 = \text{const}$$

- Shallow Water equations : no dry areas in this talk
- No local time stepping : no compensation for higher CPU time due to smaller Δts

Mathematical setting

Model equations

Seek approximate solutions of

$$\partial_t u + \nabla \cdot \boldsymbol{\mathcal{F}}(u) = \mathcal{S}(u,\, g(\vec{x})) \qquad (1)$$

on a time dependent unstructured mesh $\mathcal{T}_h(t)$.

Building blocks

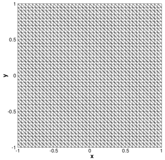
1. Discrete model for $\mathcal{T}_h(t)$: Time dependent mesh adaptation
2. Well balanced discretization of (1) on moving meshes : Well balanced ALE
3. Coupling strategy : projection and evolution or ALE ?

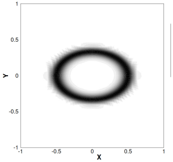
1. Time dependent mesh adaptation

- Alauzet et al *JCP* 222, 2007 :
  re-mesh and adapt to all solutions in a given time slab
- Guardone et al *JCP* 230, 2011 :
  continuous deformation with ALE and edge swap (variable topology)
- Alauzet *Eng.w.Computers* 30, 2014 :
  continuous deformations with edge swap (variable topology)
- Tang and Tang SINUM 41, 2003 :
  continuous deformation with fixed mesh topology
- etc.

Mesh adaptation by elastic deformation with fixed topology

Fixed topology : point positions change, data structure is constant ⟶ simple





HANDBOOK OF GRID GENERATION,

Thompson, Soni, and Weatherill Eds,

CRC Press, 1998

Adaptive mesh methods for one and

two dimensional hyperbolic conservation laws

H. Tang and T. Tang,

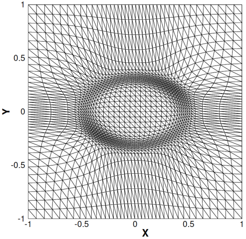
SIAM J.Numer.Anal. 2003

Adaptive moving finite volume scheme for

modelling flood inundation

F. Zhou et al.

Water Resources Reserach 2013



Time dependent mesh adaptation by elastic deformation

Elliptic "elastic" mesh movement

Given the mesh in the reference frame $\vec{X} = (X_1, X_2)$, seek $\vec{x} = \vec{x}(\vec{X})$ such that

$$\nabla_{\vec{X}} \cdot \big(\omega(\nabla_{\vec{x}} u) \, \nabla_{\vec{X}} \vec{x}\big) = \text{bc.s}$$

- Elliptic non-linear system of equations for the mapped (new) point positions $\vec{x}$, in particular (Tang and Tang, SINUM 2003) :

$$\omega(\nabla_{\vec{x}} u_h) = \sqrt{1 + \alpha \nabla u^*}\,, \quad \nabla u^* = \min\left(1, \frac{\|\nabla_{\vec{x}} u_h\|^2}{\beta^2 \max_i \|\nabla_{\vec{x}} u_i\|^2}\right)$$

Elliptic "elastic" mesh movement

Elastic analogy : setting

$$\vec{\delta} = \vec{x} - \vec{X}, \ \ \sigma = \omega \nabla_{\vec{X}} \delta, \ \ \vec{F} = -\mathbf{I}_2 \cdot \nabla_X \omega$$

we can recast last equation as

$$\nabla_X \cdot \sigma = \vec{F} + \text{bc.s}$$

- Role of $\omega = \omega(\nabla_{\vec{x}} u)$: controlling the stiffness and the force.

Elliptic "elastic" mesh movement : in practice

Elliptic PDE discretized by means of standard P^1 continuous Galerkin

$$\int_{\mathcal{T}_X} \nabla_{\vec{X}} v_h \cdot \omega(\nabla_{\vec{x}} u_h)\, \nabla_{\vec{X}} \vec{\delta}_h = \int_{\mathcal{T}_X} \mathbf{I}_2 \cdot \omega(\nabla_{\vec{x}} u_h) \nabla_X v_h + \mathsf{bc.s}$$

Leading to the non-linear system

$$\sum_j \kappa_{ij}(\vec{\delta})\vec{\delta}_j = f_i(\vec{\delta}) \quad \forall\, i$$

with $\kappa_{ij}(\delta)$ the FEM stiffness matrix and $f_i(\vec{\delta})$ the force

Elliptic "elastic" mesh movement : in practice

Solution algorithm : relaxed Newton-Jacobi iterations

$$\vec{\delta}_i^{k+1} = \vec{\delta}_i^{k} - \frac{\sum_{j\neq i} \kappa_{ij}^k \, \vec{\delta}_j^k - f_i}{\kappa_{ii}^k} \tag{2}$$

$$\vec{x}_{k+1} = \vec{x}_k + \mu \, \vec{\delta}^{k+1}$$

Important remarks

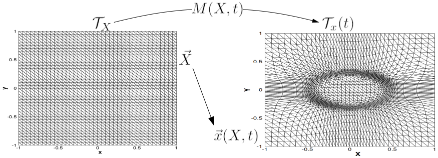
- At each iteration the FEM stiffness matrix κ_{ij}^k depends on $\nabla_{\vec{x}_k} u_h$ via ω :

  Need to compute $u_h(\vec{x}_k)$, the projection of the function u on the mesh $\vec{x}_k$

2. Well balanced schemes on moving meshes

- ref ????

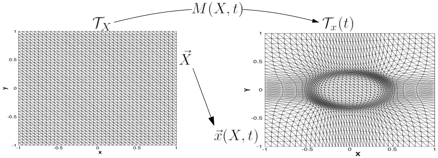
WELL BALANCED ALE



ALE RECAP FOR $\partial_t u + \nabla \cdot \boldsymbol{\mathcal{F}}(u) = 0$

- Farahat et al IJNMF 21 1995 ;
- Lesoinne and Farahat, CMAME 134, 1996 ;
- Farahat et al *JCP* 174 2001

Well balanced ALE



ALE recap for $\partial_t u + \nabla \cdot \boldsymbol{\mathcal{F}}(u) = 0$

Definitions :

Deformation speed

$$\sigma = \frac{d\vec{x}}{dt}$$

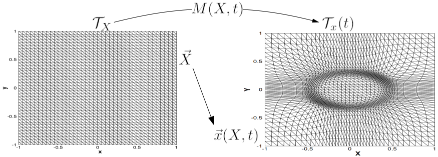
Deformation Jacobian

$$J = \det \frac{\partial \vec{x}}{\partial \vec{X}}$$

Volume :

$$V(t) = \int_{V(t)} d\vec{x} = \int_{V(t=0)} J \, d\vec{X}$$

Well balanced ALE



ALE recap for $\partial_t u + \nabla \cdot \boldsymbol{\mathcal{F}}(u) = 0$

Main results :

- Geometric Conservation Law (GCL, evolution of volume) :

$$\partial_t J\big|_{\vec{X}} = J \nabla_{\vec{x}} \cdot \sigma \tag{3}$$

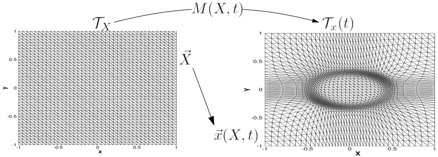
- Conservation law in ALE form (ALE-CL) :

$$\partial_t (Ju)\big|_{\vec{X}} + J\nabla_{\vec{x}} \cdot (\boldsymbol{\mathcal{F}}(u) - \sigma u) = 0 \tag{4}$$

Fundamental relation

ALE-CL reduces to GCL for constant u !!!!

Well balanced ALE



ALE recap for $\partial_t u + \nabla \cdot \boldsymbol{\mathcal{F}}(u) = 0$

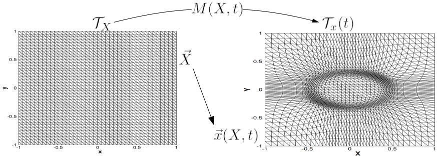
Discretization of ALE-CL, e.g. explicit FV on cell V_i :

$$V_i^{n+1} u_i^{n+1} - V_i^n u_i^n + \int_{t^n}^{t^{n+1}} \int_{\partial V_i(t)} \left(\widehat{F}(u^n) - \widehat{\sigma u}^n \right) \cdot \vec{n}(t) = 0 \qquad (5)$$

- $\widehat{F}(u)$ and $\widehat{\sigma u}$ FV numerical fluxes consistent with $\boldsymbol{\mathcal{F}}(u)$ and σu
- Discrete point diplacement speed

$$\sigma_i = \frac{\vec{x}_i^{n+1} - \vec{x}_i^n}{\Delta t} = \frac{\vec{\delta}_i}{\Delta t}$$

Well balanced ALE



ALE recap for $\partial_t u + \nabla \cdot \boldsymbol{\mathcal{F}}(u) = 0$

Discretization of ALE-CL, e.g. explicit FV on cell V_i :

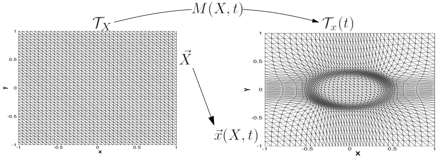
$$V_i^{n+1} u_i^{n+1} - V_i^n u_i^n + \int_{t^n}^{t^{n+1}} \int_{\partial V_i(t)} \left(\widehat{F}(u^n) - \widehat{\sigma u}^n \right) \cdot \vec{n}(t) = 0 \tag{6}$$

Fundamental relation : Discrete-GCL

To be consistent with a constant state, for $u = u_0$, the scheme MUST reduce to the identity

$$u_0 \left(V_i^{n+1} - V_i^n - \int_{t^n}^{t^{n+1}} \int_{\partial V_i(t)} \widehat{\sigma} \cdot \vec{n}(t) \right) = 0$$

Well balanced ALE



ALE recap for $\partial_t u + \nabla \cdot \boldsymbol{\mathcal{F}}(u) = 0$

Discretization of ALE-CL, e.g. explicit FV on cell V_i :

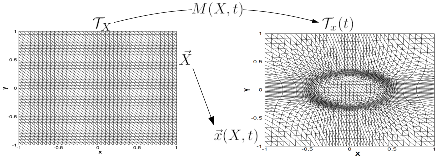
$$V_i^{n+1} u_i^{n+1} - V_i^n u_i^n + \int_{t^n}^{t^{n+1}} \int_{\partial V_i(t)} \left(\widehat{F}(u^n) - \widehat{\sigma u}^n \right) \cdot \vec{n}(t) = 0 \tag{7}$$

Fundamental relation : Discrete-GCL

Possible solution (see e.g. Farahat et al *IJNMF* 1995, for the definition of $\widehat{\sigma}$)

$$V_i^{n+1} u_i^{n+1} - V_i^n u_i^n + \Delta t \int_{\partial V_i(t^{n+1/2})} \left(\widehat{F}(u^n) - \widehat{\sigma u}^n \right) \cdot \vec{n}(t^{n+1/2}) = 0$$

Well balanced ALE



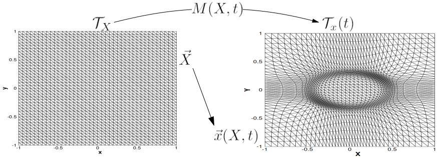
ALE for a balance law

$$\partial_t u + \nabla \cdot \boldsymbol{\mathcal{F}}(u) = \mathcal{S}(u,\, g(\vec{x}))$$

admitting a steady state characterized by

$$\eta(u,g) = \eta_0 = \text{const} \Rightarrow \nabla \cdot \boldsymbol{\mathcal{F}} = \mathcal{S}(u,\, g(\vec{x}))$$

Well balanced ALE



ALE for a balance law

$$\partial_t u + \nabla \cdot \boldsymbol{\mathcal{F}}(u) = \mathcal{S}(u,\, g(\vec{x}))$$

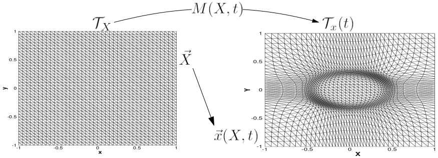
Straightforward application of ALE theory

$$\partial_t (Ju)\big|_{\vec{X}} + J\nabla_{\vec{x}} \cdot (\boldsymbol{\mathcal{F}}(u) - \sigma u) = J\mathcal{S}(u,\, g(\vec{x}))$$

plus the GCL

$$\partial_t J\big|_{\vec{X}} = J\nabla_{\vec{x}} \cdot \sigma$$

Well balanced ALE



ALE for a balance law

$$\partial_t u + \nabla \cdot \boldsymbol{\mathcal{F}}(u) = \mathcal{S}(u,\, g(\vec{x}))$$

Straightforward application of ALE theory

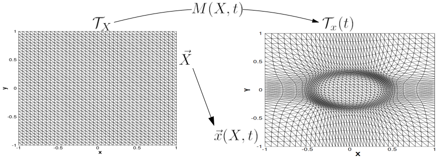
$$\partial_t (Ju)\big|_{\vec{X}} + J\nabla_{\vec{x}} \cdot (\boldsymbol{\mathcal{F}}(u) - \sigma u) = J\mathcal{S}(u,\, g(\vec{x}))$$

plus the GCL

$$\partial_t J\big|_{\vec{X}} = J\nabla_{\vec{x}} \cdot \sigma$$

Take now $\eta(u,g) = \eta_0 =$ const $\Rightarrow \nabla \cdot \boldsymbol{\mathcal{F}} = \mathcal{S}$ and combine these two relations

Well balanced ALE



ALE for a balance law

$$\partial_t u + \nabla \cdot \boldsymbol{\mathcal{F}}(u) = \mathcal{S}(u,\, g(\vec{x}))$$

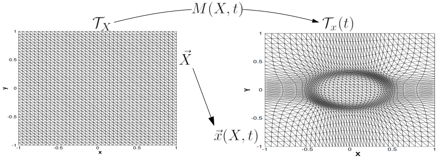
Straightforward application of ALE theory

If we take $\eta(u,g) = \eta_0 =$ const $\Rightarrow \nabla \cdot \boldsymbol{\mathcal{F}} = \mathcal{S}$ and using both relations above

$$J\partial_t u\big|_{\vec{X}} - J\sigma \cdot \nabla_{\vec{x}} u = 0$$

is this true ?

Well balanced ALE



ALE for a balance law

$$\partial_t u + \nabla \cdot \boldsymbol{\mathcal{F}}(u) = \mathcal{S}(u,\, g(\vec{x}))$$

Straightforward application of ALE theory

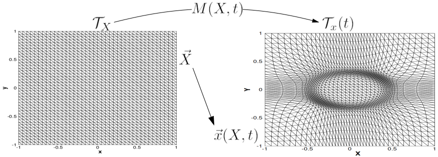
Yes (!!) since in the moving frame and for $\eta(u,g) = \eta_0 = $ const :

$$\partial_t g\big|_{\vec{X}} = \sigma \cdot \nabla_{\vec{x}} g$$

and

$$0 = \partial_t \eta\big|_{\vec{X}} - \sigma \cdot \nabla_{\vec{x}} \eta = \partial_u \eta(\partial_t u\big|_{\vec{X}} - \sigma \cdot \nabla_{\vec{x}} u) + \partial_g \eta(\partial_t g\big|_{\vec{X}} - \sigma \cdot \nabla_{\vec{x}} g)$$

Well balanced ALE



A particular case

$$\partial_t u + \nabla \cdot \boldsymbol{\mathcal{F}}(u) = \mathcal{S}(u,\, g(\vec{x}))$$

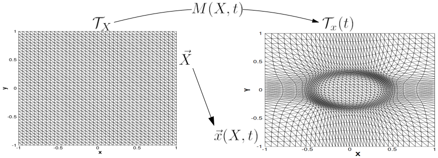
Assume that the steady balance is described by the invariant

$$\eta(u, g) = u + F(g) \Rightarrow \partial\eta = \partial u + F'(g)\partial g$$

Modified ALE form

INVENTEURS DU MONDE NUMÉRIQUE

Well balanced ALE



A particular case

$$\partial_t u + \nabla \cdot \boldsymbol{\mathcal{F}}(u) = \mathcal{S}(u,\, g(\vec{x}))$$

Assume that the steady balance is described by the invariant

$$\eta(u,g) = u + F(g) \Rightarrow \partial\eta = \partial u + F'(g)\partial g$$

Modified ALE form

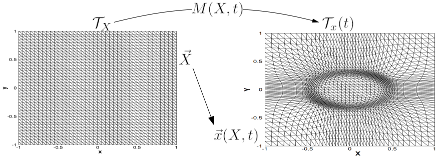
We can multiply by $F(g)$ the GCL and by $F'(g)$ the time variation of g :

$$F(g)\left(\partial_t J\big|_{\vec{X}} - J\nabla_{\vec{x}}\cdot\sigma\right) = 0 \quad \text{and} \quad F'(g)\left(J\partial_t g\big|_{\vec{X}} - J\sigma\cdot\nabla_{\vec{x}} g\right) = 0$$

and add to the std. ALE form of the balance law

$$\partial_t (Ju)\big|_{\vec{X}} + J\nabla_{\vec{x}}\cdot(\boldsymbol{\mathcal{F}}(u) - \sigma u) = J\mathcal{S}(u,\, g(\vec{x}))$$

Well balanced ALE



A particular case

$$\partial_t u + \nabla \cdot \boldsymbol{\mathcal{F}}(u) = \mathcal{S}(u,\, g(\vec{x}))$$

Assume that the steady balance is described by the invariant

$$\eta(u,g) = u + F(g) \Rightarrow \partial\eta = \partial u + F'(g)\partial g$$

Modified ALE form

Adding the resulting expressions to the original ALE form of the balance law we get

$$\partial_t(J\eta)\big|_{\vec{X}} + J\nabla_{\vec{x}} \cdot (\boldsymbol{\mathcal{F}}(u) - \sigma\eta) = J\mathcal{S}(u,\, g(\vec{x}))$$

WELL BALANCED ALE formulation

Well balanced ALE

A particular case

$$\partial_t u + \nabla \cdot \boldsymbol{\mathcal{F}}(u) = \mathcal{S}(u,\, g(\vec{x}))$$

Assume that the steady balance is described by the invariant

$$\eta(u, g) = u + F(g) \Rightarrow \partial\eta = \partial u + F'(g)\partial g$$

In summary

- Standard ALE

$$\partial_t (Ju)\big|_{\vec{X}} + J\nabla_{\vec{x}} \cdot (\boldsymbol{\mathcal{F}} - \sigma u) = J\mathcal{S}$$

- WELL BALANCED ALE

$$\partial_t (J\eta)\big|_{\vec{X}} + J\nabla_{\vec{x}} \cdot (\boldsymbol{\mathcal{F}} - \sigma \eta) = J\mathcal{S}$$

Well balanced ALE

A particular case

$$\partial_t u + \nabla \cdot \boldsymbol{\mathcal{F}}(u) = \mathcal{S}(u,\, g(\vec{x}))$$

Assume that the steady balance is described by the invariant

$$\eta(u, g) = u + F(g) \Rightarrow \partial\eta = \partial u + F'(g)\partial g$$

In summary

- Standard ALE for $\eta = u + F(g(\vec{x})) = \eta_0$

$$J(\partial_t u\big|_{\vec{X}} - \sigma \cdot \nabla_{\vec{x}} u) + u(\partial_t J\big|_{\vec{X}} - \nabla_{\vec{x}} \cdot \sigma) + J(\nabla_{\vec{x}} \boldsymbol{\mathcal{F}} - \mathcal{S}) = 0$$

- WELL BALANCED ALE for $\eta = u + F(g(\vec{x})) = \eta_0$

$$J(\partial_t \eta_0\big|_{\vec{X}} - \sigma \cdot \nabla_{\vec{x}} \eta_0) + \eta_0(\partial_t J\big|_{\vec{X}} - \nabla_{\vec{x}} \cdot \sigma) + J(\nabla_{\vec{x}} \boldsymbol{\mathcal{F}} - \mathcal{S}) = 0$$

Well balanced ALE

A particular case

$$\partial_t u + \nabla \cdot \boldsymbol{\mathcal{F}}(u) = \mathcal{S}(u,\, g(\vec{x}))$$

Assume that the steady balance is described by the invariant

$$\eta(u, g) = u + F(g) \Rightarrow \partial\eta = \partial u + F'(g)\partial g$$

In summary

- Standard ALE for $\eta = u + F(g(\vec{x})) = \eta_0$

$$J(\partial_t u\big|_{\vec{X}} - \sigma \cdot \nabla_{\vec{x}} u) + u \underbrace{(\partial_t J\big|_{\vec{X}} - \nabla_{\vec{x}} \cdot \sigma)}_{\text{DGCL}} + J \overbrace{(\nabla_{\vec{x}} \boldsymbol{\mathcal{F}} - \mathcal{S})}^{\text{Well Balanced}} = 0$$

A scheme which *verifies the DGCL*, and exactly *well balanced* on fixed meshes, will not WB be on moving meshes. The error is related to the discretization of the term

$$\partial_t u\big|_{\vec{X}} = \sigma \cdot \nabla_{\vec{x}} u$$

embedded in the scheme ...

Well balanced ALE

A particular case

$$\partial_t u + \nabla \cdot \boldsymbol{\mathcal{F}}(u) = \mathcal{S}(u,\, g(\vec{x}))$$

Assume that the steady balance is described by the invariant

$$\eta(u,g) = u + F(g) \Rightarrow \partial\eta = \partial u + F'(g)\partial g$$

In summary

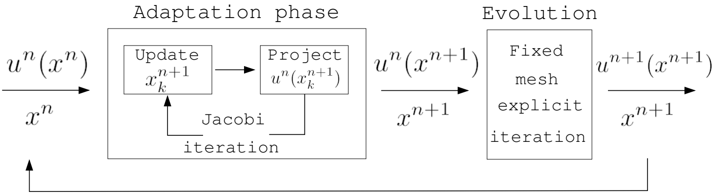
- WELL BALANCED ALE for $\eta = u + F(g(\vec{x})) = \eta_0$

$$J\overbrace{\left(\partial_t \eta_0\big|_{\vec{X}} - \sigma\cdot\nabla_{\vec{x}}\eta_0\right)}^{\partial\eta_0=0} + \eta_0\underbrace{\left(\partial_t J\big|_{\vec{X}} - \nabla_{\vec{x}}\cdot\sigma\right)}_{\text{DGCL}} + J\overbrace{\left(\nabla_{\vec{x}}\boldsymbol{\mathcal{F}} - \mathcal{S}\right)}^{\text{Well Balanced}} = 0$$

A scheme which is well balanced on fixed meshes will also be on moving meshes provided it verifies the DGCL

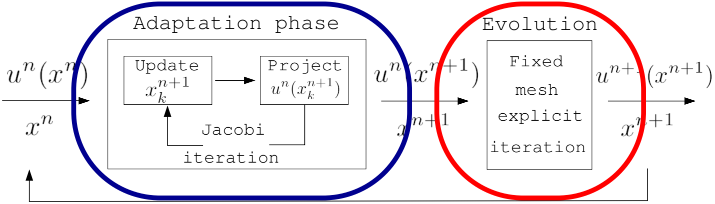
3. Adaptation-discretization coupling : projection vs ALE

DPE method



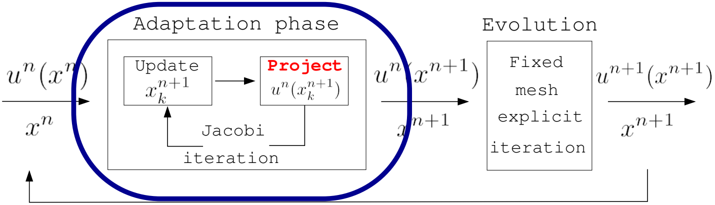
Deformation-Projection-Evolution

DPE METHOD



Deformation-Projection-Evolution

DPE method



- To get $\vec{x}_i^{n+1}$: nonlinear elliptic deformation eq. solved with initial guess $\vec{x}_i^n$
- We use 5 Jacobi iterations in all the results shown later (as suggested in Tang, Tang *SINUM* 2003)
- To compute $\omega(\nabla_{\vec{x}} u)$ we need to define a projection to get u^n onto each x_k^{n+1} (important bit)

High order conservative projection as limit of ALE

FV scheme in ALE form for a balance law

$$V_i^{n+1}\eta_i^{n+1} - V_i^n\eta_i^n + \Delta t \int_{\partial V_i(t^{n+1/2})} \left(\widehat{F}(u^n) - \widehat{\sigma\eta}^n\right) \cdot \vec{n}(t^{n+1/2}) = \Delta t V_i(t^{n+1/2})\widetilde{\mathcal{S}}_i$$

with

$$\sigma = \frac{\vec{x}^{n+1} - \vec{x}^n}{\Delta t} = \frac{\vec{\delta}}{\Delta t}$$

and $\vec{\delta}$ given from the current mesh deformation step.

High order conservative projection as limit of ALE

FV scheme in ALE form for a balance law

$$V_i^{n+1}\eta_i^{n+1} - V_i^n\eta_i^n + \Delta t \int_{\partial V_i(t^{n+1/2})} \left(\widehat{F}(u^n) - \widehat{\sigma\eta}^n\right) \cdot \vec{n}(t^{n+1/2}) = \Delta t V_i(t^{n+1/2})\widetilde{\mathcal{S}}_i$$

with

$$\sigma = \frac{\vec{x}^{n+1} - \vec{x}^n}{\Delta t} = \frac{\vec{\delta}}{\Delta t}$$

and $\vec{\delta}$ given from the current mesh deformation step.

Take now the limit for $\Delta t = 0$ and keep the displacement δ finite, moving the mesh

$$\text{from } y_k = \vec{x}_k^{n+1} \text{ to } y_{k+1} = \vec{x}_{k+1}^{n+1}$$

$$\vec{\delta} = y_{k+1} - y_k = \vec{x}_{k+1}^{n+1} - \vec{x}_k^{n+1} \neq 0$$

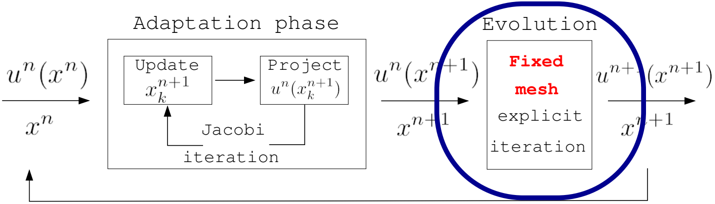
DPE method

High order conservative projection as limit of ALE

$$V_{i,k+1}\eta_i^n(y_{k+1}) - V_{i,k}\eta_i^n(y_k) - \int_{\partial V_{i,k+1/2}} \widehat{\delta\eta}^n(y_k)\cdot \vec{n}^{k+1/2} = 0$$

1. Conservative high order and well balanced projection obtained from a conservative high order well balanced scheme
2. Repeated at each Jacobi iteration : costly for high order with limiter (see next)

DPE METHOD

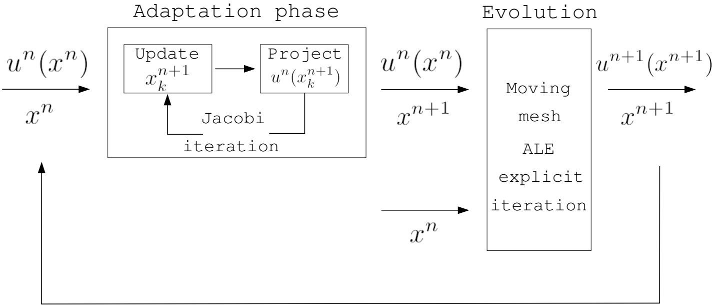


The scheme is applied on the fixed mesh as if no adaptation was used at all

1. Conservation requires the projection step needs to be conservative
2. Second order of accuracy requires the projection step to be second order accurate
3. Monotonicity requires the projection step needs to be monotone

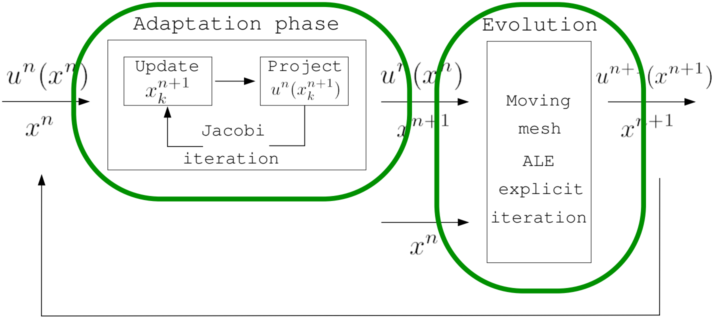
The projection step might represent a considerable cost

DALE METHOD



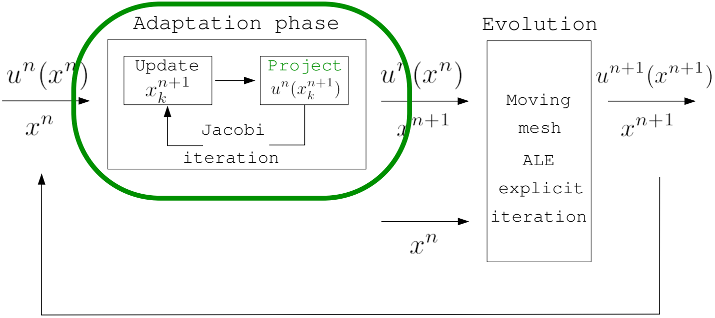
Deformation-ALE evolution

DALE METHOD



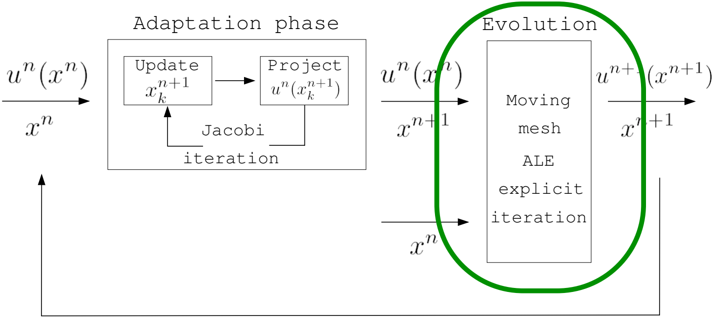
Deformation-ALE evolution

DALE METHOD



- To get $\vec{x}_i^{n+1}$: nonlinear elliptic deformation eq. solved with initial guess $\vec{x}_i^n$
- We use 5 Jacobi iterations in all the results shown later (as suggested in Tang, Tang *SINUM* 2003)
- To compute $\omega(\nabla_{\vec{x}} u)$ we need to define a projection to get u^n onto each x_k^{n+1} (important bit)

DALE method



The ALE evolution guarantees that the overall algorithm is

1. Conservative
2. Second order accurate
3. Monotone

The projection step can be simplified considerably...

Numerical examples : schemes implemented

Second order finite volume (only for scalar)

- Std well balanced Roe scheme (Bermudez-Vazquez, *Computers and Fl.* 23,1994)
- Muscl reconstruction with van Leer limiter
- Second order SSP Runge Kutta integration
- Standard ALE formulation following e.g. (Farahat et al *JCP* 174, 2001)
- Projections : zero Δt limit of first and high order Roe scheme

Second order residual distribution

- Second order positivity preserving RK-RD of (Ricchiuto and Abgrall *JCP* 2010)
- ALE extension of (Arpaia, Ricchiuto, Abgrall *JSC* 2014)
- Projections : zero Δt limit of first order Lax-Friedrich's and high order centered distributions

Scalar balance law mimicking the SW equations

$$\partial_t u + \nabla \cdot \boldsymbol{\mathcal{F}}(u) = \vec{a}(u) \cdot \nabla g(\vec{x})$$

For $\vec{a}(u) = \partial_u \boldsymbol{\mathcal{F}}$ we have a simple steady state invariant :

$$\eta = u + g(\vec{x})$$

Example 1 : linear transport with source

$$\partial_t u + \cdot\boldsymbol{\mathcal{F}} = \vec{a}\cdot\nabla g(\vec{x})$$

with

$$\vec{\boldsymbol{\mathcal{F}}} = \vec{a}u, \quad g = 0.8e^{-50(x-0.5)^2-5(y-0.9)^2)}, \quad \text{and } \vec{a}(\vec{x}) = (0,1)$$

with initial solution ($r^2 = (x-0.5)^2 + (y-0.5)^2$)

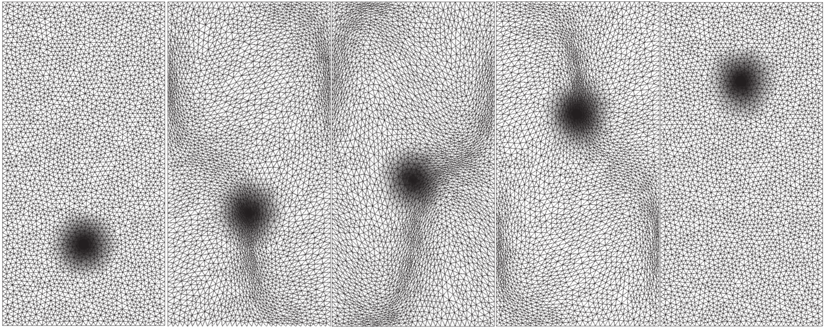
$$\eta = 1 + \psi(x,y), \quad \psi = \begin{cases} \cos^2(2\pi r) & \text{if } r < 1/4 \\ 0 & \text{otherwise} \end{cases}$$

solved on $[0,1]\times[0,2]$ superimposing the time dependent mapping

$$\begin{cases} x = X + 0.1\sin(2\pi X)\sin(\pi Y)\sin(2\pi t) \\ y = Y + 0.2\sin(2\pi X)\sin(\pi Y)\sin(4\pi t) \end{cases}$$

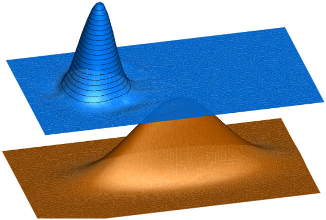
EXAMPLE 1 : LINEAR TRANSPORT WITH SOURCE

Mesh movement ($t = 0, 0.2, 0.4, 0.6, 1$)

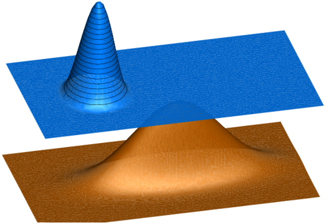


Example 1 : linear transport with source

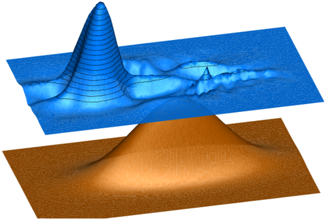
Results with linear second order RD scheme



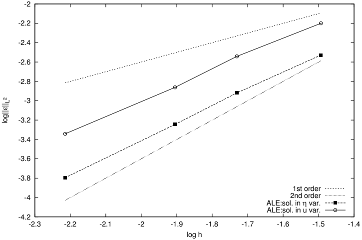
Well balanced ALE $t = 1$



Exact $t = 1$



Standard ALE $t = 1$



Grid convergence

Example 2 : rigid body rotation with source

$$\partial_t u + \cdot \boldsymbol{\mathcal{F}} = \vec{a} \cdot \nabla g(\vec{x})$$

with

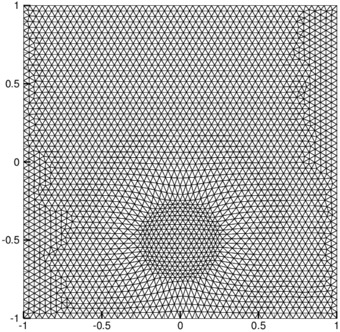
$$\vec{\boldsymbol{\mathcal{F}}} = \vec{a}(\vec{x})u, \quad g = 0.6e^{-5(x^2+y^2)}, \quad \text{and } \vec{a}(\vec{x}) = (y, -x)$$

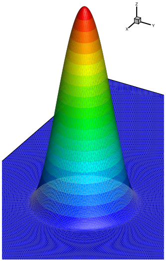
with initial solution $(r^2 = (x+0.5)^2 + y^2)$

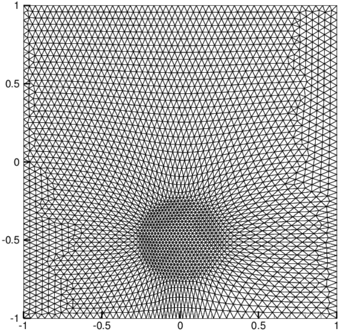
$$\eta = 1 + \psi(x,y), \quad \psi = \begin{cases} \cos^2(2\pi r) & \text{if } r < 1/4 \\ 0 & \text{otherwise} \end{cases}$$

solved on $[-1,1]^2$ testing both the DPE and DALE approaches.

Example 2 : rigid body rotation with source





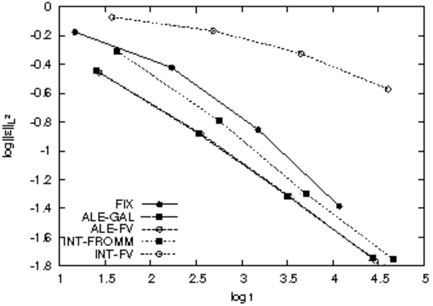


Initial

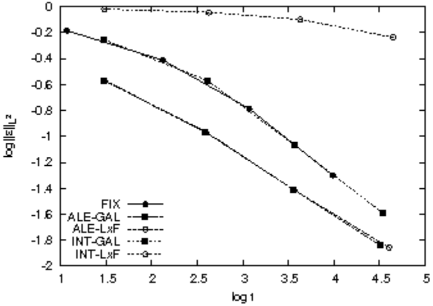
Ofter one rotation

Example 2 : rigid body rotation with source

Grid convergence : error vs CPU time



FV scheme



RD scheme

Example 3 : nonlinear balance law

$$\partial_t u + \nabla \cdot \boldsymbol{\mathcal{F}}(u) = \vec{a}(u) \cdot \nabla g(\vec{x})$$

with

$$\vec{\boldsymbol{\mathcal{F}}} = (u^2/2,\, u^2/2), \quad g = 0.6e^{-5(x^2+y^2)}, \quad \text{and } \vec{a}(u) = (u, u)$$

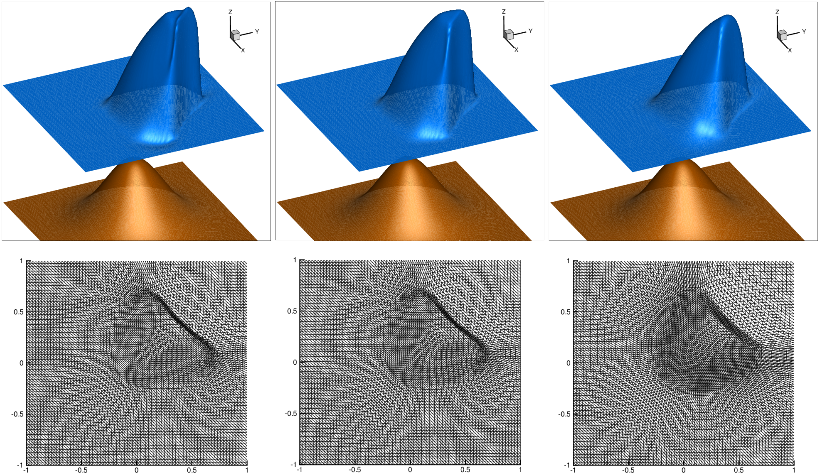
with initial solution ($r^2 = (x+0.5)^2 + y^2$)

$$\eta = 1 + \psi(x, y), \quad \psi = \begin{cases} 1.4 & \text{if } \vec{x} \in [-0.9, -0.2]^2 \\ 0.8 & \text{otherwise} \end{cases}$$

solved on $[-1, 1]^2$ testing both the DPE and DALE approaches.

Example 3 : nonlinear balance law

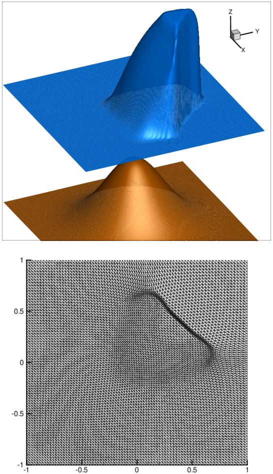
DPE results for FV



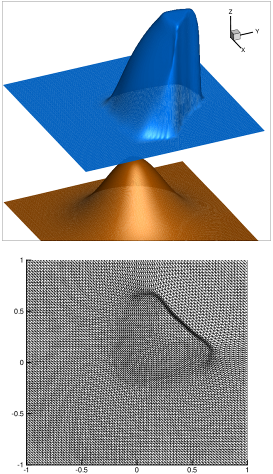
2nd order proj. High order proj. (VL limiter) 1st order proj.

Example 3 : nonlinear balance law

DALE results for FV



Simplified central 2nd order proj.

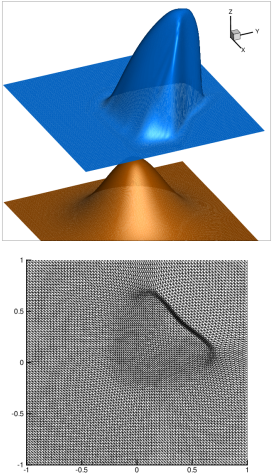


1st order proj.

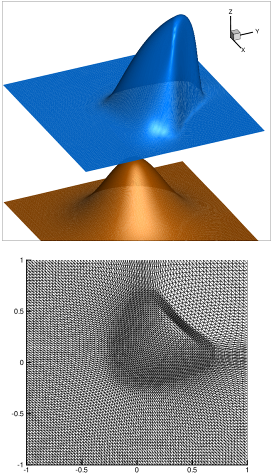
CPU gain roughly 30% w.r.t DPE

Example 3 : nonlinear balance law

DPE results for RD



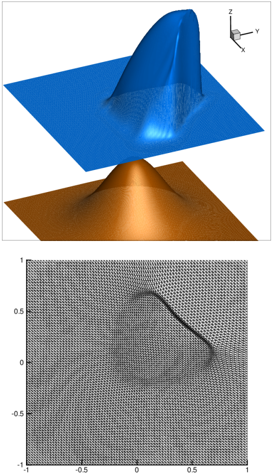
2nd order proj.



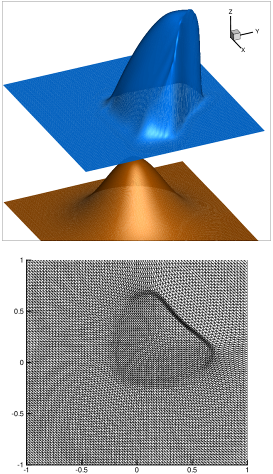
1st order proj.

Example 3 : nonlinear balance law

DALE results for RD



2nd order proj.



1st order proj.

Shallow water results with RD

Standard form

Used in the DPE algorithm

$$\partial_t \begin{bmatrix} H \\ \vec{q} \end{bmatrix} + \nabla \cdot \begin{bmatrix} \vec{q} \\ \vec{u} \otimes \vec{q} + g\dfrac{H^2}{2} \end{bmatrix} + gH \begin{bmatrix} 0 \\ \nabla b \end{bmatrix} = 0$$

Well balanced ALE form

Used in the DALE algorithm

$$\partial_t \begin{bmatrix} J\eta \\ J\vec{q} \end{bmatrix} + J\nabla \cdot \begin{bmatrix} \vec{q} - \sigma\eta \\ \vec{u} \otimes \vec{q} + g\dfrac{H^2}{2} - \sigma \otimes \vec{q} \end{bmatrix} + J\,gH \begin{bmatrix} 0 \\ \nabla b \end{bmatrix} = 0$$

Shallow water results with RD

Perturbation over smooth bathymetry

Over the domain $[0,2] \times [0,1]$ take

$$b(x,y) = 0.8e^{-50(x-0.9)^2 - 5(y-0.5)^2}$$

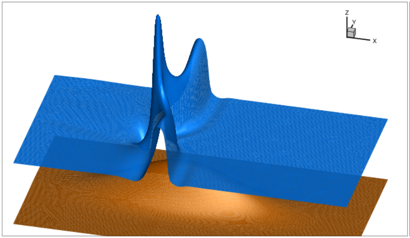
and set as initial solution still flow and free surface level

$$\eta = \begin{cases} 1.01 & \text{if } 0.05 \leq x \leq 0.15 \\ 1 & \text{otherwise} \end{cases}$$

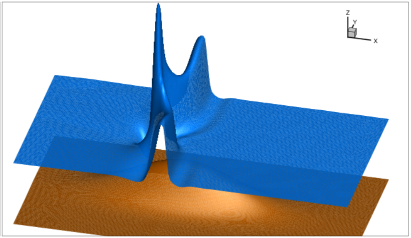
Shallow water results with RD

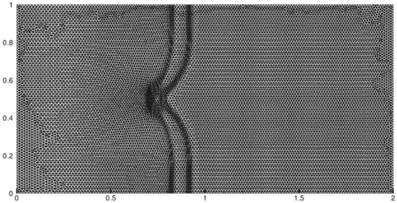
Perturbation over smooth bathymetry

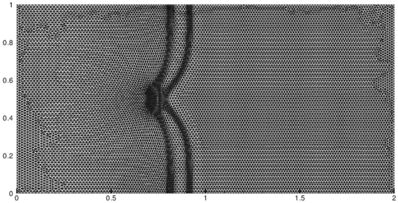
DPE with second order projection



DALE with second order projection



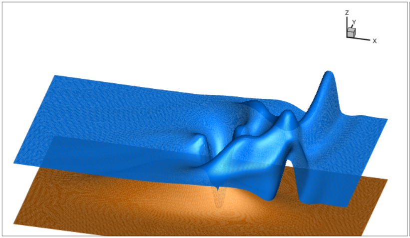




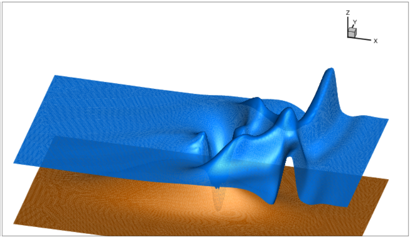
Shallow water results with RD

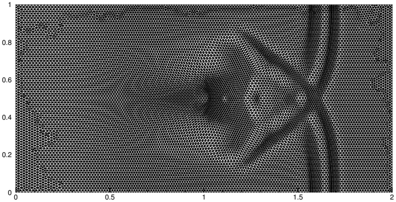
Perturbation over smooth bathymetry

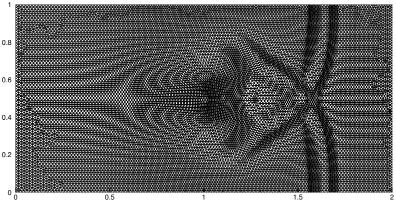
DPE with second order projection



DALE with second order projection



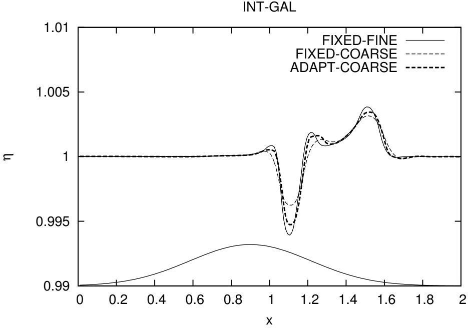


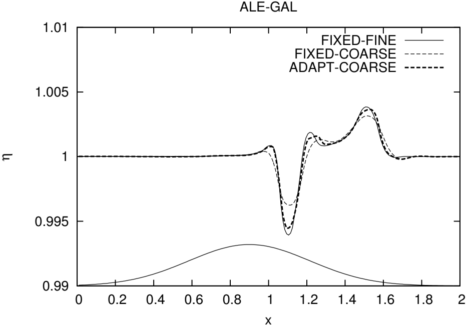


Shallow water results with RD

Perturbation over smooth bathymetry

DPE with second order projection | DALE with second order projection





CPU times :
Fixed fine : 843[s]
DPE : 246[s]
DALE : 360[s]

Dam break

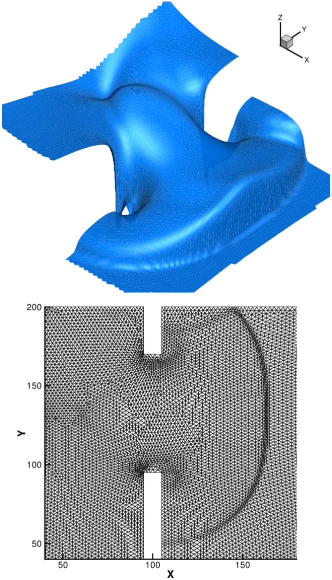
Initial solution involving still flow and

$$H_{\text{left}} = 10[m] \quad \text{and} \quad H_{\text{right}} = 5[m]$$

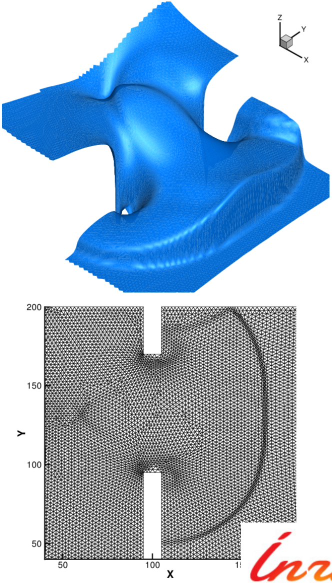
Shallow water results with RD

Dam break

DPE with second order projection



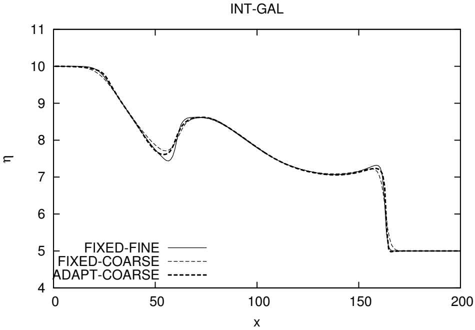
DALE with second order projection



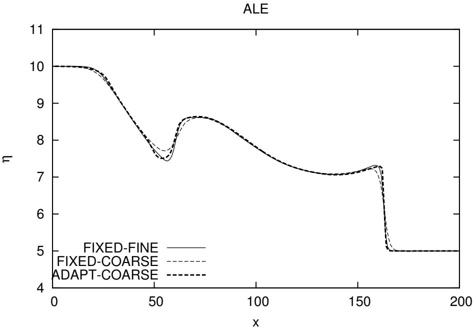
Shallow water results with RD

Dam break

DPE with second order projection



DALE with second order projection



CPU times :
Fixed fine : 220[s]
DPE : 91[s]
DALE : 97[s]

Double dam break

(Double dam break)

Conclusions and perspectives

Done so far

- Simple mesh adaptation algorithm :
  1. no major changes in code
  2. constant data structure
  3. simple point movement
  4. simple explicit Jacobi iterations for mesh adaptation
  5. need ALE formulas for projection and/or evolution
- General issue of well balanced ALE formulation
- Comparison of DPE approach and DALE approach
- DALE seems promising : better resolution/time, more flexibility

To be done

- Thorough comparison behavior of FV and RD for SW
- Dry fronts resolution (see *e.g.* Zhou et al *Water Resources Research* 2013)
- Implicit time stepping (with M.E. Hubbard)
- Improve resolution of nonlinear mesh deformation equation..
- Tsunami inundation, tidal bore formation, etc (with P. Bonneton)
- 3D and higher order schemes/curved meshes (with R. Abgrall and C. Dobrizynski)
- Local time stepping