Gradient, divergence, and laplacian discrete approximations for numerical ocean modelling

Yoann Le Bars[1,2]   Florent Lyard[3]

[1]Naval Research Laboratory, Oceanography Division
Stenniss Space Center, MS, USA

[2]University of Southern Mississippi, Department of Marine Science
Stennis Space Center, MS, USA

[3]Legos, UMR5566 CNRS-CNES-IRD-UPS
Observatoire de Midi-Pyrénées
Toulouse, France

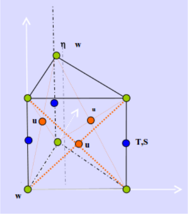
International workshop on Multiscale
(Un)-structured mesh numerical ocean Modeling
Cambridge, MA, 17-20 August 2010

T-UGOm: the Toulouse Unstructured Grid Ocean model

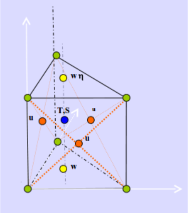
T-UGOm: 2D version use in production, experimental 3D version (research tool).

T-UGOm team:

- Florent Lyard (team leader), Legos, CNRS, Toulouse;
- Laurent Roblou, Legos, CNRS, Toulouse ;
- Yoann Le Bars, NRL, Stennis Space Center (formerly Legos);
- David Greenberg, BIO, Halifax, Nova Scottia;
- Frédéric Dupont, BIO, Halifax, Nova Scottia.



Finite element



Finite volume

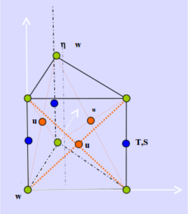
- Finite elements/volumes;
- time-splitting;
- 2D elements: triangles;
- 3D elements: prisms;
- spherical coordinates (horizontal), generalised σ (vertical);
- Boussinesq, hydrostatic;
- multiple discretisations.

T-UGOm: the Toulouse Unstructured Grid Ocean model

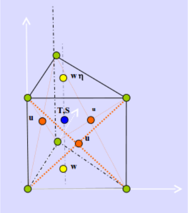
T-UGOm: 2D version use in production, experimental 3D version (research tool).

T-UGOm team:

- Florent Lyard (team leader), Legos, CNRS, Toulouse;
- Laurent Roblou, Legos, CNRS, Toulouse ;
- Yoann Le Bars, NRL, Stennis Space Center (formerly Legos);
- David Greenberg, BIO, Halifax, Nova Scottia;
- Frédéric Dupont, BIO, Halifax, Nova Scottia.



Finite element



Finite volume

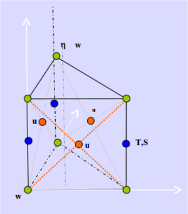
- Finite elements/volumes;
- time-splitting;
- 2D elements: triangles;
- 3D elements: prisms;
- spherical coordinates (horizontal), generalised σ (vertical);
- Boussinesq, hydrostatic;
- multiple discretisations.

T-UGOm: the Toulouse Unstructured Grid Ocean model

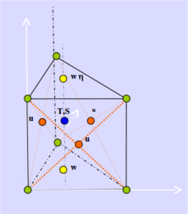
T-UGOm: 2D version use in production, experimental 3D version (research tool).

T-UGOm team:

- Florent Lyard (team leader), Legos, CNRS, Toulouse;
- Laurent Roblou, Legos, CNRS, Toulouse ;
- Yoann Le Bars, NRL, Stennis Space Center (formerly Legos);
- David Greenberg, BIO, Halifax, Nova Scottia;
- Frédéric Dupont, BIO, Halifax, Nova Scottia.



Finite element



Finite volume

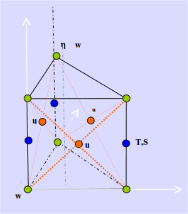
- Finite elements/volumes;
- time-splitting;
- 2D elements: triangles;
- 3D elements: prisms;
- spherical coordinates (horizontal), generalised σ (vertical);
- Boussinesq, hydrostatic;
- multiple discretisations.

T-UGOm: the Toulouse Unstructured Grid Ocean model

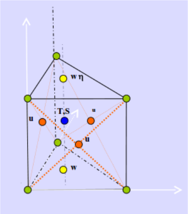
T-UGOm: 2D version use in production, experimental 3D version (research tool).

T-UGOm team:

- Florent Lyard (team leader), Legos, CNRS, Toulouse;
- Laurent Roblou, Legos, CNRS, Toulouse ;
- Yoann Le Bars, NRL, Stennis Space Center (formerly Legos);
- David Greenberg, BIO, Halifax, Nova Scottia;
- Frédéric Dupont, BIO, Halifax, Nova Scottia.



Finite element



Finite volume

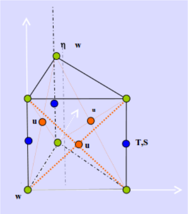
- Finite elements/volumes;
- time-splitting;
- 2D elements: triangles;
- 3D elements: prisms;
- spherical coordinates (horizontal), generalised σ (vertical);
- Boussinesq, hydrostatic;
- multiple discretisations.

T-UGOm: the Toulouse Unstructured Grid Ocean model

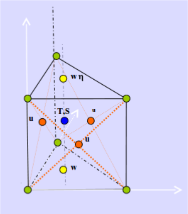
T-UGOm: 2D version use in production, experimental 3D version (research tool).

T-UGOm team:

- Florent Lyard (team leader), Legos, CNRS, Toulouse;
- Laurent Roblou, Legos, CNRS, Toulouse ;
- Yoann Le Bars, NRL, Stennis Space Center (formerly Legos);
- David Greenberg, BIO, Halifax, Nova Scottia;
- Frédéric Dupont, BIO, Halifax, Nova Scottia.



Finite element



Finite volume

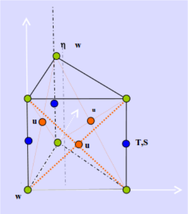
- Finite elements/volumes;
- time-splitting;
- 2D elements: triangles;
- 3D elements: prisms;
- spherical coordinates (horizontal), generalised σ (vertical);
- Boussinesq, hydrostatic;
- multiple discretisations.

T-UGOm: the Toulouse Unstructured Grid Ocean model

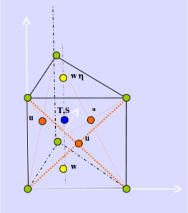
T-UGOm: 2D version use in production, experimental 3D version (research tool).

T-UGOm team:

- Florent Lyard (team leader), Legos, CNRS, Toulouse;
- Laurent Roblou, Legos, CNRS, Toulouse ;
- Yoann Le Bars, NRL, Stennis Space Center (formerly Legos);
- David Greenberg, BIO, Halifax, Nova Scottia;
- Frédéric Dupont, BIO, Halifax, Nova Scottia.



Finite element



Finite volume

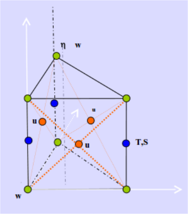
- Finite elements/volumes;
- time-splitting;
- 2D elements: triangles;
- 3D elements: prisms;
- spherical coordinates (horizontal), generalised σ (vertical);
- Boussinesq, hydrostatic;
- multiple discretisations.

T-UGOm: the Toulouse Unstructured Grid Ocean model

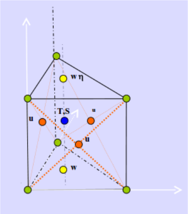
T-UGOm: 2D version use in production, experimental 3D version (research tool).

T-UGOm team:

- Florent Lyard (team leader), Legos, CNRS, Toulouse;
- Laurent Roblou, Legos, CNRS, Toulouse ;
- Yoann Le Bars, NRL, Stennis Space Center (formerly Legos);
- David Greenberg, BIO, Halifax, Nova Scottia;
- Frédéric Dupont, BIO, Halifax, Nova Scottia.



Finite element



Finite volume

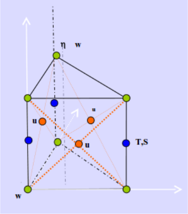
- Finite elements/volumes;
- time-splitting;
- 2D elements: triangles;
- 3D elements: prisms;
- spherical coordinates (horizontal), generalised σ (vertical);
- Boussinesq, hydrostatic;
- multiple discretisations.

T-UGOm: the Toulouse Unstructured Grid Ocean model

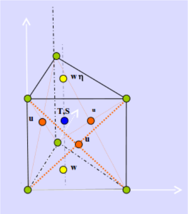
T-UGOm: 2D version use in production, experimental 3D version (research tool).

T-UGOm team:

- Florent Lyard (team leader), Legos, CNRS, Toulouse;
- Laurent Roblou, Legos, CNRS, Toulouse ;
- Yoann Le Bars, NRL, Stennis Space Center (formerly Legos);
- David Greenberg, BIO, Halifax, Nova Scottia;
- Frédéric Dupont, BIO, Halifax, Nova Scottia.



Finite element



Finite volume

- Finite elements/volumes;
- time-splitting;
- 2D elements: triangles;
- 3D elements: prisms;
- spherical coordinates (horizontal), generalised σ (vertical);
- Boussinesq, hydrostatic;
- multiple discretisations.

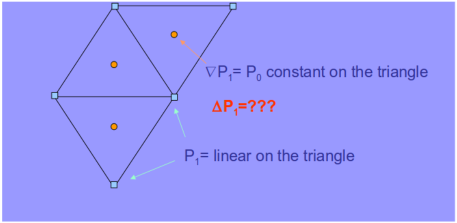
Finite elements versus finites differences

Purpose:

1. Deeper understanding of similarities and differences between finite differences, finite elements and finite volumes;
2. make a rigorous generalised formalism, that can be used either in continuous and discontinuous case.

- Finite elements: explicit projection, derivation of interpolation functions (continuous interpolation);
- finite differences: implicit projection, rate of increase (discontinuous interpolation).

Not so different:



Generalised formalism

Finite elements "contain" finite differences

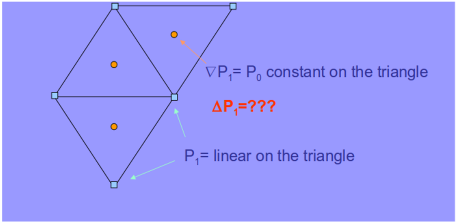
Finite elements versus finites differences

Purpose:

1. Deeper understanding of similarities and differences between finite differences, finite elements and finite volumes;
2. make a rigorous generalised formalism, that can be used either in continuous and discontinuous case.

- Finite elements: explicit projection, derivation of interpolation functions (continuous interpolation);
- finite differences: implicit projection, rate of increase (discontinuous interpolation).

Not so different:



Generalised formalism

Finite elements "contain" finite differences

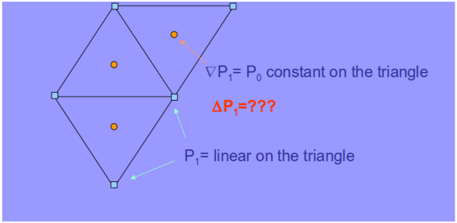
Finite elements versus finites differences

Purpose:

1. Deeper understanding of similarities and differences between finite differences, finite elements and finite volumes;
2. make a rigorous generalised formalism, that can be used either in continuous and discontinuous case.

- Finite elements: explicit projection, derivation of interpolation functions (continuous interpolation);
- finite differences: implicit projection, rate of increase (discontinuous interpolation).

Not so different:



Generalised formalism

Finite elements “contain” finite differences

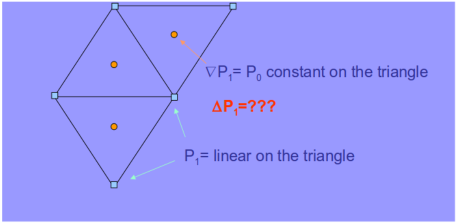
Finite elements versus finites differences

Purpose:

1. Deeper understanding of similarities and differences between finite differences, finite elements and finite volumes;
2. make a rigorous generalised formalism, that can be used either in continuous and discontinuous case.

- Finite elements: explicit projection, derivation of interpolation functions (continuous interpolation);
- finite differences: implicit projection, rate of increase (discontinuous interpolation).

Not so different:



Generalised formalism

Finite elements “contain” finite differences

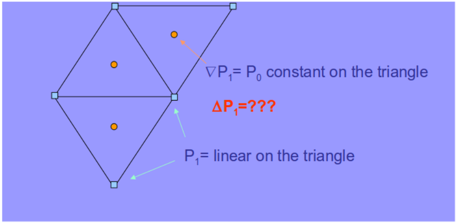
Finite elements versus finites differences

Purpose:

1. Deeper understanding of similarities and differences between finite differences, finite elements and finite volumes;
2. make a rigorous generalised formalism, that can be used either in continuous and discontinuous case.

- Finite elements: explicit projection, derivation of interpolation functions (continuous interpolation);
- finite differences: implicit projection, rate of increase (discontinuous interpolation).

Not so different:



Generalised formalism

Finite elements “contain” finite differences

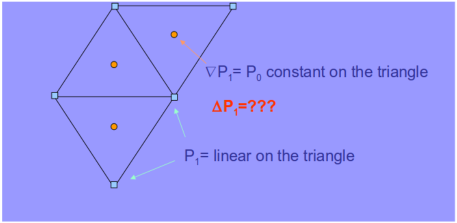
Finite elements versus finites differences

Purpose:

1. Deeper understanding of similarities and differences between finite differences, finite elements and finite volumes;
2. make a rigorous generalised formalism, that can be used either in continuous and discontinuous case.

- Finite elements: explicit projection, derivation of interpolation functions (continuous interpolation);
- finite differences: implicit projection, rate of increase (discontinuous interpolation).

Not so different:



Generalised formalism

Finite elements “contain” finite differences

Some difficulties

- Shallow water and generalised wave equations: $\boldsymbol{\nabla} \cdot \overline{\boldsymbol{u}}$ and $\boldsymbol{\nabla}\eta$
  $\Rightarrow$ need to compute discrete divergences and gradients;
- advection-diffusion equation: $\nabla \cdot \nabla c$ (laplacian can be also use for kinetic momentum equation)
  $\Rightarrow$ need to compute laplacians of discontinuous functions (non-measurable);
- σ-layers modelling: discontinuous elevation
  $\Rightarrow$ discontinuous σ-layer discretisation, commonly simply ignored
  $\Rightarrow$ hydrostatic inconstancy.

Some difficulties

- Shallow water and generalised wave equations: $\nabla \cdot \bar{\boldsymbol{u}}$ and $\nabla \eta$
  $\Rightarrow$ need to compute discrete divergences and gradients;
- advection-diffusion equation: $\nabla \cdot \nabla c$ (laplacian can be also use for kinetic momentum equation)
  $\Rightarrow$ need to compute laplacians of discontinuous functions (non-measurable);
- σ-layers modelling: discontinuous elevation
  $\Rightarrow$ discontinuous σ-layer discretisation, commonly simply ignored
  $\Rightarrow$ hydrostatic inconstancy.

Some difficulties

- Shallow water and generalised wave equations: $\boldsymbol{\nabla} \cdot \bar{\boldsymbol{u}}$ and $\boldsymbol{\nabla}\eta$
  $\Rightarrow$ need to compute discrete divergences and gradients;
- advection-diffusion equation: $\boldsymbol{\nabla} \cdot \boldsymbol{\nabla} c$ (laplacian can be also use for kinetic momentum equation)
  $\Rightarrow$ need to compute laplacians of discontinuous functions (non-measurable);
- σ-layers modelling: discontinuous elevation
  $\Rightarrow$ discontinuous σ-layer discretisation, commonly simply ignored
  $\Rightarrow$ hydrostatic inconstancy.

Some difficulties

- Shallow water and generalised wave equations: $\nabla \cdot \bar{\boldsymbol{u}}$ and $\nabla \eta$
  $\Rightarrow$ need to compute discrete divergences and gradients;
- advection-diffusion equation: $\nabla \cdot \nabla c$ (laplacian can be also use for kinetic momentum equation)
  $\Rightarrow$ need to compute laplacians of discontinuous functions (non-measurable);
- σ-layers modelling: discontinuous elevation
  $\Rightarrow$ discontinuous σ-layer discretisation, commonly simply ignored
  $\Rightarrow$ hydrostatic inconstancy.

Some difficulties

- Shallow water and generalised wave equations: $\nabla \cdot \bar{\boldsymbol{u}}$ and $\nabla \eta$
  $\Rightarrow$ need to compute discrete divergences and gradients;
- advection-diffusion equation: $\nabla \cdot \nabla c$ (laplacian can be also use for kinetic momentum equation)
  $\Rightarrow$ need to compute laplacians of discontinuous functions (non-measurable);
- σ-layers modelling: discontinuous elevation
  $\Rightarrow$ discontinuous σ-layer discretisation, commonly simply ignored
  $\Rightarrow$ hydrostatic inconstancy.

Some difficulties

- Shallow water and generalised wave equations: $\nabla \cdot \bar{\boldsymbol{u}}$ and $\nabla \eta$
  $\Rightarrow$ need to compute discrete divergences and gradients;
- advection-diffusion equation: $\nabla \cdot \nabla c$ (laplacian can be also use for kinetic momentum equation)
  $\Rightarrow$ need to compute laplacians of discontinuous functions (non-measurable);
- σ-layers modelling: discontinuous elevation
  $\Rightarrow$ discontinuous σ-layer discretisation, commonly simply ignored
  $\Rightarrow$ hydrostatic inconstancy.

Some difficulties

- Shallow water and generalised wave equations: $\nabla \cdot \bar{\boldsymbol{u}}$ and $\nabla \eta$
  $\Rightarrow$ need to compute discrete divergences and gradients;
- advection-diffusion equation: $\nabla \cdot \nabla c$ (laplacian can be also use for kinetic momentum equation)
  $\Rightarrow$ need to compute laplacians of discontinuous functions (non-measurable);
- σ-layers modelling: discontinuous elevation
  $\Rightarrow$ discontinuous σ-layer discretisation, commonly simply ignored
  $\Rightarrow$ hydrostatic inconstancy.

Outlook

1. Divergence and gradient
2. Looking for an optimal reformulation of laplacian in discontinuous case
3. σ-layers modelling

Outlook

1. Divergence and gradient

2. Looking for an optimal reformulation of laplacian in discontinuous case

3. σ-layers modelling

Definitions

Measures

Definition

$\forall E$ *measurable space,* $\forall m \in \mathbb{N}^\star$ *and* $p \in [1;+\infty[$*,* $W^{m,p}(E)$*: Sobolev's space.*

Definition

$d\boldsymbol{x}$*: Lebesgue's measure.*

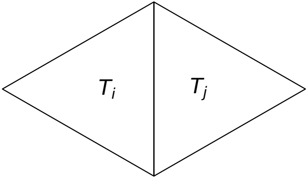
Definition

$d\sigma(\boldsymbol{x})$*: frontier measure (e.g. 2D: path length).*

Definitions

Notations

Let T_i and T_j be two neighbouring elements and $\Gamma_{i,j}$ their common edge.



$\forall \psi \in W^{m,p}(E)$ and, $\forall n \in \mathbb{N}^{\star}$, $\forall \boldsymbol{f} \in \left[W^{m,p}(E)\right]^n$, let:

$$\overline{\overline{\psi}}_{i,j} = \frac{1}{2}\left(\psi|_{T_i} + \psi|_{T_j}\right), \qquad (1)$$

$$\overline{\overline{\boldsymbol{f}}}_{i,j} = \frac{1}{2}\left(\boldsymbol{f}|_{T_i} + \boldsymbol{f}|_{T_j}\right). \qquad (2)$$

$\forall \beta_{i,j} \in [0;1]$ such as $\beta_{j,i} = 1 - \beta_{i,j}$, let:

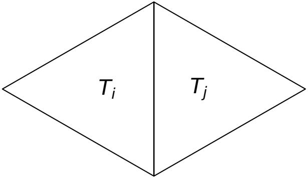
$$\underline{\underline{\psi}}_{i,j} = \beta_{i,j}\left(\psi|_{T_j} - \psi|_{T_i}\right), \qquad (3)$$

$$\underline{\underline{\boldsymbol{f}}}_{i,j} = \beta_{i,j}\left(\boldsymbol{f}|_{T_j} - \boldsymbol{f}|_{T_i}\right). \qquad (4)$$

Definitions

Notations

Let T_i and T_j be two neighbouring elements and $\Gamma_{i,j}$ their common edge.



$\forall \psi \in W^{m,p}(E)$ and, $\forall n \in \mathbb{N}^\star$, $\forall \boldsymbol{f} \in \left[W^{m,p}(E)\right]^n$, let:

$$\overline{\overline{\psi}}_{i,j} = \frac{1}{2}\left(\psi|_{T_i} + \psi|_{T_j}\right), \qquad (1)$$

$$\overline{\overline{\boldsymbol{f}}}_{i,j} = \frac{1}{2}\left(\boldsymbol{f}|_{T_i} + \boldsymbol{f}|_{T_j}\right). \qquad (2)$$

$\forall \beta_{i,j} \in [0;1]$ such as $\beta_{j,i} = 1 - \beta_{i,j}$, let:

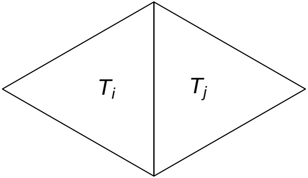
$$\underline{\underline{\psi}}_{i,j} = \beta_{i,j}\left(\psi|_{T_j} - \psi|_{T_i}\right), \qquad (3)$$

$$\underline{\underline{\boldsymbol{f}}}_{i,j} = \beta_{i,j}\left(\boldsymbol{f}|_{T_j} - \boldsymbol{f}|_{T_i}\right). \qquad (4)$$

Definitions

Notations

Let T_i and T_j be two neighbouring elements and $\Gamma_{i,j}$ their common edge.



$\forall \psi \in W^{m,p}\left(E\right)$ and, $\forall n \in \mathbb{N}^{\star}$, $\forall \boldsymbol{f} \in \left[W^{m,p}\left(E\right)\right]^n$, let:

$$\overline{\overline{\psi}}_{i,j} = \frac{1}{2}\left(\psi|_{T_i} + \psi|_{T_j}\right), \tag{1}$$

$$\overline{\overline{\boldsymbol{f}}}_{i,j} = \frac{1}{2}\left(\boldsymbol{f}|_{T_i} + \boldsymbol{f}|_{T_j}\right). \tag{2}$$

$\forall \beta_{i,j} \in \left[0;1\right]$ such as $\beta_{j,i} = 1 - \beta_{i,j}$, let:

$$\underline{\underline{\psi}}_{i,j} = \beta_{i,j}\left(\psi|_{T_j} - \psi|_{T_i}\right), \tag{3}$$

$$\underline{\underline{\boldsymbol{f}}}_{i,j} = \beta_{i,j}\left(\boldsymbol{f}|_{T_j} - \boldsymbol{f}|_{T_i}\right). \tag{4}$$

Proposed approximations

Basics

Analogy with Dirac's measure

$$\int_{\Gamma_{i,j}} \boldsymbol{\nabla} \psi d\sigma\left(\boldsymbol{x}\right) \approx \int_{\Gamma_{i,j}} \underline{\underline{\psi}}_{i,j} \cdot \boldsymbol{n}_{i,j} d\sigma\left(\boldsymbol{x}\right), \quad (5)$$

$$\int_{\Gamma_{i,j}} \boldsymbol{\nabla} \cdot \boldsymbol{f} d\sigma\left(\boldsymbol{x}\right) \approx \int_{\Gamma_{i,j}} \underline{\underline{\boldsymbol{f}}}_{i,j} \cdot \boldsymbol{n}_{i,j} d\sigma\left(\boldsymbol{x}\right), \quad (6)$$

$\boldsymbol{n}_{i,j}$: the unit vector normal to $\Gamma_{i,j}$ and orientated from T_i to T_j.

Proposed approximations

Actual approximation

$$\int_{\Omega_k} \psi \boldsymbol{\nabla} \cdot \boldsymbol{f} d\boldsymbol{x} = \sum_{i=1}^{N_k} \int_{\mathring{T}_i} \psi \boldsymbol{\nabla} \cdot \boldsymbol{f} d\boldsymbol{x} + \sum_{i=1}^{N_k} \sum_{j : T_j \in \{\text{neighbours of } T_i\}} \int_{\Gamma_{i,j}} \psi \boldsymbol{\nabla} \cdot \boldsymbol{f} d\sigma(\boldsymbol{x}), \quad (7)$$

Ω_k: discretised problem space,

N_k: number of elements,

$\mathring{T}_i$: interior of T_i.

Analogy with finite differences

$$\int_{\Gamma_{i,j}} \psi \boldsymbol{\nabla} \cdot \boldsymbol{f} d\sigma(\boldsymbol{x}) \approx \int_{\Gamma_{i,j}} \overline{\overline{\psi}}_{i,j} \underline{\underline{\boldsymbol{f}}}_{i,j} \cdot \boldsymbol{n}_{i,j} d\sigma(\boldsymbol{x}). \quad (8)$$

Proposed approximations

Actual approximation

$$\int_{\Omega_k} \psi \boldsymbol{\nabla} \cdot \boldsymbol{f} d\boldsymbol{x} = \sum_{i=1}^{N_k} \int_{\mathring{T}_i} \psi \boldsymbol{\nabla} \cdot \boldsymbol{f} d\boldsymbol{x} + \underbrace{\sum_{i=1}^{N_k} \sum_{j:T_j \in \{\text{neighbours of } T_i\}} \int_{\Gamma_{i,j}} \psi \boldsymbol{\nabla} \cdot \boldsymbol{f} d\sigma(\boldsymbol{x})}_{\text{discontinuity}}, \quad (7)$$

Ω_k: discretised problem space,

N_k: number of elements,

$\mathring{T}_i$: interior of T_i.

Analogy with finite differences

$$\int_{\Gamma_{i,j}} \psi \boldsymbol{\nabla} \cdot \boldsymbol{f} d\sigma(\boldsymbol{x}) \approx \int_{\Gamma_{i,j}} \overline{\overline{\psi}}_{i,j} \underline{\underline{\boldsymbol{f}}}_{i,j} \cdot \boldsymbol{n}_{i,j} d\sigma(\boldsymbol{x}). \quad (8)$$

Proposed approximations

Actual approximation

$$\int_{\Omega_k} \psi \boldsymbol{\nabla}\cdot\boldsymbol{f} d\boldsymbol{x} = \sum_{i=1}^{N_k} \int_{\mathring{T}_i} \psi \boldsymbol{\nabla}\cdot\boldsymbol{f} d\boldsymbol{x} + \underbrace{\sum_{i=1}^{N_k} \sum_{j:T_j \in \{\text{neighbours of } T_i\}} \int_{\Gamma_{i,j}} \psi \boldsymbol{\nabla}\cdot\boldsymbol{f} d\sigma(\boldsymbol{x})}_{\text{discontinuity}}, \quad (7)$$

Ω_k: discretised problem space,

N_k: number of elements,

$\mathring{T}_i$: interior of T_i.

Analogy with finite differences

$$\int_{\Gamma_{i,j}} \psi \boldsymbol{\nabla}\cdot\boldsymbol{f} d\sigma(\boldsymbol{x}) \approx \int_{\Gamma_{i,j}} \overline{\overline{\psi}}_{i,j} \underline{\underline{\boldsymbol{f}}}_{i,j}\cdot\boldsymbol{n}_{i,j} d\sigma(\boldsymbol{x}). \quad (8)$$

Standard integration properties

Proposition (Leibniz's formula)

$\forall E$ measurable space, either continuous or discrete, with the proposed approximations, the following Leibniz's formula is verified:

$$\int_E \boldsymbol{\nabla} \cdot \psi \boldsymbol{f} d\boldsymbol{x} = \int_E \psi \boldsymbol{\nabla} \cdot \boldsymbol{f} d\boldsymbol{x} + \int_E \boldsymbol{\nabla} \psi \cdot \boldsymbol{f} d\boldsymbol{x}.$$

Proposition (Stokes' formula)

Let Γ_E be E fronter and $\boldsymbol{n}_{\Gamma_E}(\boldsymbol{x})$ be the unit vector normal to Γ_E on point $\boldsymbol{x}$, directed to the outside. Then, with the proposed approximations,
$\forall E$ measurable space, either continuous or discrete, Stokes' formula is verified:

$$\int_E \boldsymbol{\nabla} \cdot \boldsymbol{f} d\boldsymbol{x} = \int_{\Gamma_E} \boldsymbol{f} \cdot \boldsymbol{n}_{\Gamma_E} d\sigma(\boldsymbol{x}).$$

(Le Bars 2010; Lyard and Le Bars in prep.)

Outlook

1 Divergence and gradient

2 Looking for an optimal reformulation of laplacian in discontinuous case

3 σ-layers modelling

Objectives

1. $\forall \xi \in W^{m,p}(\Omega_k)$, to determine values of $\int_{\Omega_k} \boldsymbol{\nabla} \cdot \psi \boldsymbol{\nabla} \xi d\boldsymbol{x}$ and $\int_{\Omega_k} \boldsymbol{\nabla} \psi \cdot \boldsymbol{\nabla} \xi d\boldsymbol{x}$:
   - two derivations, then one integration;
   - only one derivation in previous gradient and divergence definitions;
2. to verify the following two Leibniz's formulas:

$$\int_{\Omega_k} \boldsymbol{\nabla} \cdot \boldsymbol{\nabla} (\psi \xi) d\boldsymbol{x} = \int_{\Omega_k} \psi \boldsymbol{\nabla} \cdot \boldsymbol{\nabla} \xi d\boldsymbol{x} + 2 \int_{\Omega_k} \boldsymbol{\nabla} \xi \cdot \boldsymbol{\nabla} \psi d\boldsymbol{x} + \int_{\Omega_k} \xi \boldsymbol{\nabla} \cdot \boldsymbol{\nabla} \psi d\boldsymbol{x}, \quad (9)$$

$$\int_{\Omega_k} \boldsymbol{\nabla} \cdot \xi \boldsymbol{\nabla} \psi d\boldsymbol{x} = \int_{\Omega_k} \xi \boldsymbol{\nabla} \cdot \boldsymbol{\nabla} \psi d\boldsymbol{x} + \int_{\Omega_k} \boldsymbol{\nabla} \xi \cdot \boldsymbol{\nabla} \psi d\boldsymbol{x}; \quad (10)$$

3. with Λ an arbitrary discontinuity repartition factor, to verify Stokes' formula:

$$\int_{\Omega_k} \boldsymbol{\nabla} \cdot \boldsymbol{\nabla} \psi = \sum_{i=1}^{N_k} \sum_{j:T_j \in \{\text{neighbours of } T_i\}} \left(\int_{\Gamma_{i,j}} \boldsymbol{n}_{i,j} \cdot \boldsymbol{\nabla} \psi d\sigma(\boldsymbol{x}) + \int_{\Gamma_{i,j}} \Lambda \boldsymbol{n}_{i,j} \cdot \underline{\underline{\boldsymbol{\nabla} \psi}}_{i,j} d\sigma(\boldsymbol{x}) \right) \quad (11)$$

Objectives

1. $\forall \xi \in W^{m,p}(\Omega_k)$, to determine values of $\int_{\Omega_k} \boldsymbol{\nabla} \cdot \psi \boldsymbol{\nabla} \xi d\boldsymbol{x}$ and $\int_{\Omega_k} \boldsymbol{\nabla} \psi \cdot \boldsymbol{\nabla} \xi d\boldsymbol{x}$:
   - two derivations, then one integration;
   - only one derivation in previous gradient and divergence definitions;
2. to verify the following two Leibniz's formulas:

$$\int_{\Omega_k} \boldsymbol{\nabla} \cdot \boldsymbol{\nabla} (\psi \xi) \, d\boldsymbol{x} = \int_{\Omega_k} \psi \boldsymbol{\nabla} \cdot \boldsymbol{\nabla} \xi d\boldsymbol{x} + 2 \int_{\Omega_k} \boldsymbol{\nabla} \xi \cdot \boldsymbol{\nabla} \psi d\boldsymbol{x} + \int_{\Omega_k} \xi \boldsymbol{\nabla} \cdot \boldsymbol{\nabla} \psi d\boldsymbol{x}, \tag{9}$$

$$\int_{\Omega_k} \boldsymbol{\nabla} \cdot \xi \boldsymbol{\nabla} \psi d\boldsymbol{x} = \int_{\Omega_k} \xi \boldsymbol{\nabla} \cdot \boldsymbol{\nabla} \psi d\boldsymbol{x} + \int_{\Omega_k} \boldsymbol{\nabla} \xi \cdot \boldsymbol{\nabla} \psi d\boldsymbol{x}; \tag{10}$$

3. with Λ an arbitrary discontinuity repartition factor, to verify Stokes' formula:

$$\int_{\Omega_k} \boldsymbol{\nabla} \cdot \boldsymbol{\nabla} \psi = \sum_{i=1}^{N_k} \sum_{j:T_j \in \{\text{neighbours of } T_i\}} \left(\int_{\Gamma_{i,j}} \boldsymbol{n}_{i,j} \cdot \boldsymbol{\nabla} \psi d\sigma(\boldsymbol{x}) + \int_{\Gamma_{i,j}} \Lambda \boldsymbol{n}_{i,j} \cdot \underline{\underline{\boldsymbol{\nabla} \psi}}_{i,j} d\sigma(\boldsymbol{x}) \right) \tag{11}$$

Objectives

1. $\forall \xi \in W^{m,p}(\Omega_k)$, to determine values of $\int_{\Omega_k} \boldsymbol{\nabla} \cdot \psi \boldsymbol{\nabla} \xi d\boldsymbol{x}$ and $\int_{\Omega_k} \boldsymbol{\nabla} \psi \cdot \boldsymbol{\nabla} \xi d\boldsymbol{x}$:
   - two derivations, then one integration;
   - only one derivation in previous gradient and divergence definitions;
2. to verify the following two Leibniz's formulas:

$$\int_{\Omega_k} \boldsymbol{\nabla} \cdot \boldsymbol{\nabla}(\psi\xi)\, d\boldsymbol{x} = \int_{\Omega_k} \psi \boldsymbol{\nabla} \cdot \boldsymbol{\nabla} \xi d\boldsymbol{x} + 2\int_{\Omega_k} \boldsymbol{\nabla} \xi \cdot \boldsymbol{\nabla} \psi d\boldsymbol{x} + \int_{\Omega_k} \xi \boldsymbol{\nabla} \cdot \boldsymbol{\nabla} \psi d\boldsymbol{x}, \quad (9)$$

$$\int_{\Omega_k} \boldsymbol{\nabla} \cdot \xi \boldsymbol{\nabla} \psi d\boldsymbol{x} = \int_{\Omega_k} \xi \boldsymbol{\nabla} \cdot \boldsymbol{\nabla} \psi d\boldsymbol{x} + \int_{\Omega_k} \boldsymbol{\nabla} \xi \cdot \boldsymbol{\nabla} \psi d\boldsymbol{x}; \quad (10)$$

3. with Λ an arbitrary discontinuity repartition factor, to verify Stokes' formula:

$$\int_{\Omega_k} \boldsymbol{\nabla} \cdot \boldsymbol{\nabla} \psi = \sum_{i=1}^{N_k} \sum_{j:T_j \in \{\text{neighbours of } T_i\}} \left(\int_{\Gamma_{i,j}} \boldsymbol{n}_{i,j} \cdot \boldsymbol{\nabla} \psi d\sigma(\boldsymbol{x}) + \int_{\Gamma_{i,j}} \Lambda \boldsymbol{n}_{i,j} \cdot \underline{\underline{\boldsymbol{\nabla} \psi}}_{i,j} d\sigma(\boldsymbol{x}) \right) \quad (11)$$

Objectives

1. $\forall \xi \in W^{m,p}(\Omega_k)$, to determine values of $\int_{\Omega_k} \boldsymbol{\nabla} \cdot \psi \boldsymbol{\nabla} \xi d\boldsymbol{x}$ and $\int_{\Omega_k} \boldsymbol{\nabla} \psi \cdot \boldsymbol{\nabla} \xi d\boldsymbol{x}$:
   - two derivations, then one integration;
   - only one derivation in previous gradient and divergence definitions;
2. to verify the following two Leibniz's formulas:

$$\int_{\Omega_k} \boldsymbol{\nabla} \cdot \boldsymbol{\nabla} (\psi \xi)\, d\boldsymbol{x} = \int_{\Omega_k} \psi \boldsymbol{\nabla} \cdot \boldsymbol{\nabla} \xi d\boldsymbol{x} + 2 \int_{\Omega_k} \boldsymbol{\nabla} \xi \cdot \boldsymbol{\nabla} \psi d\boldsymbol{x} + \int_{\Omega_k} \xi \boldsymbol{\nabla} \cdot \boldsymbol{\nabla} \psi d\boldsymbol{x}, \quad (9)$$

$$\int_{\Omega_k} \boldsymbol{\nabla} \cdot \xi \boldsymbol{\nabla} \psi d\boldsymbol{x} = \int_{\Omega_k} \xi \boldsymbol{\nabla} \cdot \boldsymbol{\nabla} \psi d\boldsymbol{x} + \int_{\Omega_k} \boldsymbol{\nabla} \xi \cdot \boldsymbol{\nabla} \psi d\boldsymbol{x}; \quad (10)$$

3. with Λ an arbitrary discontinuity repartition factor, to verify Stokes' formula:

$$\int_{\Omega_k} \boldsymbol{\nabla} \cdot \boldsymbol{\nabla} \psi = \sum_{i=1}^{N_k} \sum_{j:T_j \in \{\text{neighbours of } T_i\}} \left(\int_{\Gamma_{i,j}} \boldsymbol{n}_{i,j} \cdot \boldsymbol{\nabla} \psi d\sigma(\boldsymbol{x}) + \int_{\Gamma_{i,j}} \Lambda \boldsymbol{n}_{i,j} \cdot \underline{\underline{\boldsymbol{\nabla} \psi}}_{i,j} d\sigma(\boldsymbol{x}) \right) \quad (11)$$

Objectives

1. $\forall \xi \in W^{m,p}(\Omega_k)$, to determine values of $\int_{\Omega_k} \boldsymbol{\nabla} \cdot \psi \boldsymbol{\nabla} \xi d\boldsymbol{x}$ and $\int_{\Omega_k} \boldsymbol{\nabla} \psi \cdot \boldsymbol{\nabla} \xi d\boldsymbol{x}$:
   - two derivations, then one integration;
   - only one derivation in previous gradient and divergence definitions;
2. to verify the following two Leibniz's formulas:

$$\int_{\Omega_k} \boldsymbol{\nabla} \cdot \boldsymbol{\nabla} (\psi \xi)\, d\boldsymbol{x} = \int_{\Omega_k} \psi \boldsymbol{\nabla} \cdot \boldsymbol{\nabla} \xi d\boldsymbol{x} + 2 \int_{\Omega_k} \boldsymbol{\nabla} \xi \cdot \boldsymbol{\nabla} \psi d\boldsymbol{x} + \int_{\Omega_k} \xi \boldsymbol{\nabla} \cdot \boldsymbol{\nabla} \psi d\boldsymbol{x}, \tag{9}$$

$$\int_{\Omega_k} \boldsymbol{\nabla} \cdot \xi \boldsymbol{\nabla} \psi d\boldsymbol{x} = \int_{\Omega_k} \xi \boldsymbol{\nabla} \cdot \boldsymbol{\nabla} \psi d\boldsymbol{x} + \int_{\Omega_k} \boldsymbol{\nabla} \xi \cdot \boldsymbol{\nabla} \psi d\boldsymbol{x}; \tag{10}$$

3. with Λ an arbitrary discontinuity repartition factor, to verify Stokes' formula:

$$\int_{\Omega_k} \boldsymbol{\nabla} \cdot \boldsymbol{\nabla} \psi = \sum_{i=1}^{N_k} \sum_{j:T_j \in \{\text{neighbours of } T_i\}} \left(\int_{\Gamma_{i,j}} \boldsymbol{n}_{i,j} \cdot \boldsymbol{\nabla} \psi d\sigma(\boldsymbol{x}) + \int_{\Gamma_{i,j}} \Lambda \boldsymbol{n}_{i,j} \cdot \underline{\underline{\boldsymbol{\nabla} \psi}}_{i,j} d\sigma(\boldsymbol{x}) \right) \tag{11}$$

Methodology

For now on, we see three possibilities:

1. to define a bounded discontinuous gradient, then use previous divergence definition;
2. to determine continuous field approximation and then compute laplacian;
3. use a filter to convolute the discontinuous field, then the gradient and so the laplacian can be controlled with filter slope (not detailed here).

Methodology

For now on, we see three possibilities:

1. to define a bounded discontinuous gradient, then use previous divergence definition;
2. to determine continuous field approximation and then compute laplacian;
3. use a filter to convolute the discontinuous field, then the gradient and so the laplacian can be controlled with filter slope (not detailed here).

Methodology

For now on, we see three possibilities:

1. to define a bounded discontinuous gradient, then use previous divergence definition;
2. to determine continuous field approximation and then compute laplacian;
3. use a filter to convolute the discontinuous field, then the gradient and so the laplacian can be controlled with filter slope (not detailed here).

Methodology

For now on, we see three possibilities:

1. to define a bounded discontinuous gradient, then use previous divergence definition;
2. to determine continuous field approximation and then compute laplacian;
3. use a filter to convolute the discontinuous field, then the gradient and so the laplacian can be controlled with filter slope (not detailed here).

Using previous developments

Bounded discontinuous gradient, then use previous divergence definition

$$
\begin{aligned}
\int_{\Omega_k} \xi \boldsymbol{\nabla} \cdot \boldsymbol{\nabla} \psi d\boldsymbol{x} = \sum_{i=1}^{N_k} \Bigg[\int_{\overset{\circ}{T_i}} \xi \boldsymbol{\nabla} \cdot \boldsymbol{\nabla} \psi d\boldsymbol{x} + & \sum_{j:T_j \in \{\text{neighbours of } T_i\}} \left(\int_{\Gamma_{i,j}} \Lambda \overline{\overline{\xi}}_{i,j} \underline{\underline{\boldsymbol{\nabla} \psi}}_{i,j} \cdot \boldsymbol{n}_{i,j} d\sigma(\boldsymbol{x}) \right. \\
& - \int_{\Gamma_{i,j}} \Lambda \boldsymbol{n}_{i,j} \cdot \underline{\underline{\psi}}_{i,j} \overline{\overline{\boldsymbol{\nabla} \xi}}_{i,j} d\sigma(\boldsymbol{x}) + \frac{L_e}{A_e} \int_{\Gamma_{i,j}} \Lambda \overline{\overline{\xi}}_{i,j} \underline{\underline{\psi}}_{i,j} d\sigma(\boldsymbol{x}) \\
& \left. - \frac{L_e}{A_e} \int_{\Gamma_{i,j}} \Lambda^2 \underline{\underline{\xi}}_{i,j} \times \underline{\underline{\psi}}_{i,j} d\sigma(\boldsymbol{x}) \right) \Bigg],
\end{aligned}
\tag{12}
$$

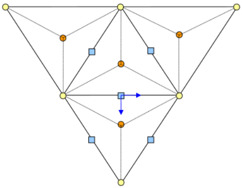
$$
\begin{aligned}
\int_{\Omega_k} \boldsymbol{\nabla} \cdot \xi \boldsymbol{\nabla} \psi d\boldsymbol{x} = \sum_{i=1}^{N_k} \Bigg[\int_{\overset{\circ}{T_i}} \boldsymbol{\nabla} \cdot \xi \boldsymbol{\nabla} \psi d\boldsymbol{x} + & \sum_{j:T_j \in \{\text{neighbours of } T_i\}} \left(\int_{\Gamma_{i,j}} \Lambda \underline{\underline{\xi \boldsymbol{\nabla} \psi}}_{i,j} \cdot \boldsymbol{n}_{i,j} d\sigma(\boldsymbol{x}) \right. \\
& \left. + \frac{L_e}{A_e} \int_{\Gamma_{i,j}} \Lambda \overline{\overline{\xi}}_{i,j} \underline{\underline{\psi}}_{i,j} d\sigma(\boldsymbol{x}) \right) \Bigg]
\end{aligned}
\tag{13}
$$

L_e: edge measurement (e.g 2D: segment length),
A_e: element measurement (e.g. 2D: element area).

(Lyard and Le Bars in prep.)

Continuous field approximation

$$\forall i \in \{1,2,\dots,N_k\}\,,\ \boldsymbol{\nabla}\psi|_{T_i} = \boldsymbol{\nabla}\psi|_{\overset{\circ}{T_i}} + \sum_{j:T_j\in\{\text{neighbours of } T_i\}} \underline{\underline{\psi}}_{i,j}\,\delta_{i,j}\boldsymbol{n}_{i,j}$$



Analogy with least squares method: determine $\boldsymbol{u}$ constant on T_i such as

$$\int_{T_i} \left\| \boldsymbol{\nabla}\psi - \boldsymbol{\nabla}\psi|_{\overset{\circ}{T_i}} - \boldsymbol{u} \right\|^2 d\boldsymbol{x} =$$

$$\sum_{j:T_j\in\{\text{neighbours of } T_i\}} \int_{\Gamma_{i,j}} \left\| \underline{\underline{\psi}}_{i,j}\,\delta_{i,j}\boldsymbol{n}_{i,j} - \boldsymbol{u} \right\|^2 d\sigma(\boldsymbol{x})$$

Laplacian of the continuous approximation

$$\int_{\Omega_k} \boldsymbol{\nabla}\cdot\xi\boldsymbol{\nabla}\psi d\boldsymbol{x} = \sum_{i=1}^{N_k}\int_{\overset{\circ}{T_i}} \boldsymbol{\nabla}\cdot\xi\boldsymbol{\nabla}\psi d\boldsymbol{x} + \sum_{i=1}^{N_k}\sum_{j:T_j\in\{\text{neighbours of } T_i\}}\int_{\Gamma_{i,j}} \underline{\underline{\xi\boldsymbol{\nabla}\psi}}_{i,j}\cdot\boldsymbol{n}_{i,j}d\sigma(\boldsymbol{x}) \quad (14)$$

According to Stokes' formula:

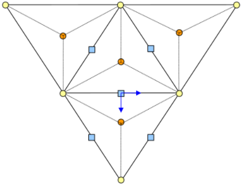
$$\int_{\Omega_k} \xi\boldsymbol{\nabla}\cdot\boldsymbol{\nabla}\psi d\boldsymbol{x} = \sum_{i=1}^{N_k}\Bigg[\int_{\overset{\circ}{T_i}} \xi\boldsymbol{\nabla}\cdot\boldsymbol{\nabla}\psi d\boldsymbol{x} + \sum_{j:T_j\in\{\text{neighbours of } T_i\}}\Bigg(\int_{\Gamma_{i,j}}\Lambda\overline{\overline{\xi}}_{i,j}\underline{\underline{\boldsymbol{\nabla}\psi}}_{i,j}\cdot n_{i,j}d\sigma(\boldsymbol{x}) \quad (15)$$

$$- \int_{\Gamma_{i,j}}\Lambda\boldsymbol{n}_{i,j}\cdot\underline{\underline{\psi}}_{i,j}\overline{\overline{\boldsymbol{\nabla}\xi}}_{i,j}d\sigma(\boldsymbol{x}) - \frac{1}{A_e}\int_{\Gamma_{i,j}}\Lambda\underline{\underline{\psi}}_{i,j}d\sigma(\boldsymbol{x})\int_{\Gamma_{i,j}}\Lambda\underline{\underline{\xi}}_{i,j}d\sigma(\boldsymbol{x})\Bigg)\Bigg]$$

(Lyard and Le Bars in prep.)

Continuous field approximation

$$\forall i \in \{1,2,\dots,N_k\}\,,\ \boldsymbol{\nabla}\psi|_{T_i} = \boldsymbol{\nabla}\psi|_{\mathring{T}_i} + \sum_{j:T_j\in\{\text{neighbours of } T_i\}} \underline{\underline{\psi}}_{i,j}\,\delta_{i,j}\boldsymbol{n}_{i,j}$$



Analogy with least squares method: determine $\boldsymbol{u}$ constant on T_i such as

$$\int_{T_i} \left\| \boldsymbol{\nabla}\psi - \boldsymbol{\nabla}\psi|_{\mathring{T}_i} - \boldsymbol{u} \right\|^2 d\boldsymbol{x} =$$

$$\sum_{j:T_j\in\{\text{neighbours of } T_i\}} \int_{\Gamma_{i,j}} \left\| \underline{\underline{\psi}}_{i,j}\,\delta_{i,j}\boldsymbol{n}_{i,j} - \boldsymbol{u} \right\|^2 d\sigma(\boldsymbol{x})$$

Laplacian of the continuous approximation

$$\int_{\Omega_k} \nabla\cdot\xi\boldsymbol{\nabla}\psi d\boldsymbol{x} = \sum_{i=1}^{N_k}\int_{\mathring{T}_i} \nabla\cdot\xi\boldsymbol{\nabla}\psi d\boldsymbol{x} + \sum_{i=1}^{N_k}\sum_{j:T_j\in\{\text{neighbours of } T_i\}}\int_{\Gamma_{i,j}} \underline{\underline{\xi\boldsymbol{\nabla}\psi}}_{i,j}\cdot\boldsymbol{n}_{i,j}d\sigma(\boldsymbol{x}) \qquad (14)$$

According to Stokes' formula:

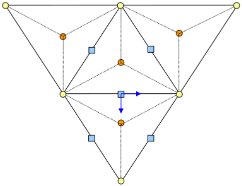
$$\int_{\Omega_k}\xi\boldsymbol{\nabla}\cdot\boldsymbol{\nabla}\psi d\boldsymbol{x} = \sum_{i=1}^{N_k}\Bigg[\int_{\mathring{T}_i}\xi\boldsymbol{\nabla}\cdot\boldsymbol{\nabla}\psi d\boldsymbol{x} + \sum_{j:T_j\in\{\text{neighbours of } T_i\}}\Bigg(\int_{\Gamma_{i,j}}\Lambda\overline{\overline{\xi}}_{i,j}\underline{\underline{\boldsymbol{\nabla}\psi}}_{i,j}\cdot n_{i,j}d\sigma(\boldsymbol{x}) \qquad (15)$$

$$-\int_{\Gamma_{i,j}}\Lambda\boldsymbol{n}_{i,j}\cdot\underline{\underline{\psi}}_{i,j}\overline{\overline{\boldsymbol{\nabla}\xi}}_{i,j}d\sigma(\boldsymbol{x}) - \frac{1}{A_e}\int_{\Gamma_{i,j}}\Lambda\underline{\underline{\psi}}_{i,j}d\sigma(\boldsymbol{x})\int_{\Gamma_{i,j}}\Lambda\underline{\underline{\xi}}_{i,j}d\sigma(\boldsymbol{x})\Bigg)\Bigg]$$

(Lyard and Le Bars in prep.)

Continuous field approximation

$$\forall i \in \{1, 2, \ldots, N_k\}\,,\ \boldsymbol{\nabla}\psi|_{T_i} = \boldsymbol{\nabla}\psi|_{\overset{\circ}{T}_i} + \sum_{j:T_j \in \{\text{neighbours of } T_i\}} \underline{\underline{\psi}}_{i,j}\, \delta_{i,j} \boldsymbol{n}_{i,j}$$



Analogy with least squares method: determine $\boldsymbol{u}$ constant on T_i such as

$$\int_{T_i} \left\| \boldsymbol{\nabla}\psi - \boldsymbol{\nabla}\psi|_{\overset{\circ}{T}_i} - \boldsymbol{u} \right\|^2 d\boldsymbol{x} =$$

$$\sum_{j:T_j \in \{\text{neighbours of } T_i\}} \int_{\Gamma_{i,j}} \left\| \underline{\underline{\psi}}_{i,j}\, \delta_{i,j} \boldsymbol{n}_{i,j} - \boldsymbol{u} \right\|^2 d\sigma(\boldsymbol{x})$$

Laplacian of the continuous approximation

$$\int_{\Omega_k} \boldsymbol{\nabla} \cdot \xi \boldsymbol{\nabla}\psi d\boldsymbol{x} = \sum_{i=1}^{N_k} \int_{\overset{\circ}{T}_i} \boldsymbol{\nabla} \cdot \xi \boldsymbol{\nabla}\psi d\boldsymbol{x} + \sum_{i=1}^{N_k} \sum_{j:T_j \in \{\text{neighbours of } T_i\}} \int_{\Gamma_{i,j}} \underline{\underline{\xi \boldsymbol{\nabla}\psi}}_{i,j} \cdot \boldsymbol{n}_{i,j} d\sigma(\boldsymbol{x}) \qquad (14)$$

According to Stokes' formula:

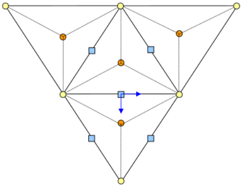
$$\int_{\Omega_k} \xi \boldsymbol{\nabla} \cdot \boldsymbol{\nabla}\psi d\boldsymbol{x} = \sum_{i=1}^{N_k} \Bigg[\int_{\overset{\circ}{T}_i} \xi \boldsymbol{\nabla} \cdot \boldsymbol{\nabla}\psi d\boldsymbol{x} + \sum_{j:T_j \in \{\text{neighbours of } T_i\}} \Bigg(\int_{\Gamma_{i,j}} \Lambda \overline{\overline{\xi}}_{i,j} \underline{\underline{\boldsymbol{\nabla}\psi}}_{i,j} \cdot n_{i,j} d\sigma(\boldsymbol{x}) \qquad (15)$$

$$- \int_{\Gamma_{i,j}} \Lambda \boldsymbol{n}_{i,j} \cdot \underline{\underline{\psi}}_{i,j} \overline{\overline{\boldsymbol{\nabla}\xi}}_{i,j} d\sigma(\boldsymbol{x}) - \frac{1}{A_e} \int_{\Gamma_{i,j}} \Lambda \underline{\underline{\psi}}_{i,j} d\sigma(\boldsymbol{x}) \int_{\Gamma_{i,j}} \Lambda \underline{\underline{\xi}}_{i,j} d\sigma(\boldsymbol{x}) \Bigg) \Bigg]$$

(Lyard and Le Bars in prep.)

Continuous field approximation

$$\forall i \in \{1,2,\ldots,N_k\}, \ \nabla \psi|_{T_i} = \nabla \psi|_{\overset{\circ}{T_i}} + \sum_{j:T_j \in \{\text{neighbours of } T_i\}} \underline{\underline{\psi}}_{i,j} \delta_{i,j} \boldsymbol{n}_{i,j}$$



Analogy with least squares method: determine $\boldsymbol{u}$ constant on T_i such as

$$\int_{T_i} \left\| \nabla \psi - \nabla \psi|_{\overset{\circ}{T_i}} - \boldsymbol{u} \right\|^2 d\boldsymbol{x} =$$

$$\sum_{j:T_j \in \{\text{neighbours of } T_i\}} \int_{\Gamma_{i,j}} \left\| \underline{\underline{\psi}}_{i,j} \delta_{i,j} \boldsymbol{n}_{i,j} - \boldsymbol{u} \right\|^2 d\sigma(\boldsymbol{x})$$

Laplacian of the continuous approximation

$$\int_{\Omega_k} \nabla \cdot \xi \nabla \psi d\boldsymbol{x} = \sum_{i=1}^{N_k} \int_{\overset{\circ}{T_i}} \nabla \cdot \xi \nabla \psi d\boldsymbol{x} + \sum_{i=1}^{N_k} \sum_{j:T_j \in \{\text{neighbours of } T_i\}} \int_{\Gamma_{i,j}} \underline{\underline{\xi \nabla \psi}}_{i,j} \cdot \boldsymbol{n}_{i,j} d\sigma(\boldsymbol{x}) \quad (14)$$

According to Stokes' formula:

$$\int_{\Omega_k} \xi \nabla \cdot \nabla \psi d\boldsymbol{x} = \sum_{i=1}^{N_k} \Bigg[\int_{\overset{\circ}{T_i}} \xi \nabla \cdot \nabla \psi d\boldsymbol{x} + \sum_{j:T_j \in \{\text{neighbours of } T_i\}} \Bigg(\int_{\Gamma_{i,j}} \Lambda \overline{\overline{\xi}}_{i,j} \underline{\underline{\nabla \psi}}_{i,j} \cdot n_{i,j} d\sigma(\boldsymbol{x}) \quad (15)$$

$$- \int_{\Gamma_{i,j}} \Lambda \boldsymbol{n}_{i,j} \cdot \underline{\underline{\psi}}_{i,j} \overline{\overline{\nabla \xi}}_{i,j} d\sigma(\boldsymbol{x}) - \frac{1}{A_e} \int_{\Gamma_{i,j}} \Lambda \underline{\underline{\psi}}_{i,j} d\sigma(\boldsymbol{x}) \int_{\Gamma_{i,j}} \Lambda \underline{\underline{\xi}}_{i,j} d\sigma(\boldsymbol{x}) \Bigg) \Bigg]$$

(Lyard and Le Bars in prep.)

Outlook

3D σ-layers, finite volumes formulation

Finite differences equations (change of vertical coordinates) can be seen as layer-integrated equations, with application of Leibniz's formula

Volume conservation: $0 = \int_{s_0}^{s_1} \nabla \cdot \boldsymbol{u} ds = \int_{s_0}^{s_1} \nabla_H \cdot \boldsymbol{v} dz + [w]_{s_0}^{s_1}$

According to Leibniz's formula: $\int_{s_0}^{s_1} \nabla \cdot \boldsymbol{u} ds = \nabla_H \cdot \int_{s_0}^{s_1} \boldsymbol{v} dz - [\boldsymbol{v} \cdot \nabla_H s]$

$\boldsymbol{u}$: 3D velocity;
$\boldsymbol{v}$: horizontal velocity;
w: vertical velocity;
ω: omega velocity.

Then:

$$\int_{s_0}^{s_1} \nabla \cdot \boldsymbol{u} ds = \nabla_H \cdot \int_{s_0}^{s_1} \boldsymbol{v} dz + [\omega]_{s_0}^{s_1} + \left[\frac{\partial s}{\partial t}\right]_{s_0}^{s_1} \qquad (16)$$

In case of element-wise constant elevation/layer position:

- spatial derivations require correction terms due to discontinuities;
- Leibniz's formula does not hold (i.e. requires additional correction terms) if layers are not face to face from one column to another;
- *most known models do not care properly about those two issues, may it be because it would significantly increase computation time?*

3D σ-layers, finite volumes formulation

Finite differences equations (change of vertical coordinates) can be seen as layer-integrated equations, with application of Leibniz's formula

Volume conservation: $0 = \int_{s_0}^{s_1} \boldsymbol{\nabla} \cdot \boldsymbol{u} ds = \int_{s_0}^{s_1} \boldsymbol{\nabla}_H \cdot \boldsymbol{v} dz + [w]_{s_0}^{s_1}$

According to Leibniz's formula: $\int_{s_0}^{s_1} \boldsymbol{\nabla} \cdot \boldsymbol{u} ds = \boldsymbol{\nabla}_H \cdot \int_{s_0}^{s_1} \boldsymbol{v} dz - [\boldsymbol{v} \cdot \boldsymbol{\nabla}_H s]$

Then: $$\int_{s_0}^{s_1} \boldsymbol{\nabla} \cdot \boldsymbol{u} ds = \boldsymbol{\nabla}_H \cdot \int_{s_0}^{s_1} \boldsymbol{v} dz + [\omega]_{s_0}^{s_1} + \left[\frac{\partial s}{\partial t}\right]_{s_0}^{s_1} \tag{16}$$

$\boldsymbol{u}$: 3D velocity;
$\boldsymbol{v}$: horizontal velocity;
w: vertical velocity;
ω: omega velocity.

In case of element-wise constant elevation/layer position:

- spatial derivations require correction terms due to discontinuities;
- Leibniz's formula does not hold (i.e. requires additional correction terms) if layers are not face to face from one column to another;
- *most known models do not care properly about those two issues, may it be because it would significantly increase computation time?*

3D σ-layers, finite volumes formulation

Finite differences equations (change of vertical coordinates) can be seen as layer-integrated equations, with application of Leibniz's formula

Volume conservation: $0 = \int_{s_0}^{s_1} \boldsymbol{\nabla} \cdot \boldsymbol{u} ds = \int_{s_0}^{s_1} \boldsymbol{\nabla}_H \cdot \boldsymbol{v} dz + [w]_{s_0}^{s_1}$

According to Leibniz's formula: $\int_{s_0}^{s_1} \boldsymbol{\nabla} \cdot \boldsymbol{u} ds = \boldsymbol{\nabla}_H \cdot \int_{s_0}^{s_1} \boldsymbol{v} dz - [\boldsymbol{v} \cdot \boldsymbol{\nabla}_H s]$

$\boldsymbol{u}$: 3D velocity;
$\boldsymbol{v}$: horizontal velocity;
w: vertical velocity;
ω: omega velocity.

Then: $$\int_{s_0}^{s_1} \boldsymbol{\nabla} \cdot \boldsymbol{u} ds = \boldsymbol{\nabla}_H \cdot \int_{s_0}^{s_1} \boldsymbol{v} dz + [\omega]_{s_0}^{s_1} + \left[\frac{\partial s}{\partial t}\right]_{s_0}^{s_1} \quad (16)$$

In case of element-wise constant elevation/layer position:

- spatial derivations require correction terms due to discontinuities;
- Leibniz's formula does not hold (i.e. requires additional correction terms) if layers are not face to face from one column to another;
- *most known models do not care properly about those two issues, may it be because it would significantly increase computation time?*

3D σ-layers, finite volumes formulation

Finite differences equations (change of vertical coordinates) can be seen as layer-integrated equations, with application of Leibniz's formula

Volume conservation: $0 = \int_{s_0}^{s_1} \boldsymbol{\nabla} \cdot \boldsymbol{u} ds = \int_{s_0}^{s_1} \boldsymbol{\nabla}_H \cdot \boldsymbol{v} dz + [w]_{s_0}^{s_1}$

According to Leibniz's formula: $\int_{s_0}^{s_1} \boldsymbol{\nabla} \cdot \boldsymbol{u} ds = \boldsymbol{\nabla}_H \cdot \int_{s_0}^{s_1} \boldsymbol{v} dz - [\boldsymbol{v} \cdot \boldsymbol{\nabla}_H s]$

$$\text{Then: } \int_{s_0}^{s_1} \boldsymbol{\nabla} \cdot \boldsymbol{u} ds = \boldsymbol{\nabla}_H \cdot \int_{s_0}^{s_1} \boldsymbol{v} dz + [\omega]_{s_0}^{s_1} + \left[\frac{\partial s}{\partial t}\right]_{s_0}^{s_1} \qquad (16)$$

$\boldsymbol{u}$: 3D velocity;
$\boldsymbol{v}$: horizontal velocity;
w: vertical velocity;
ω: omega velocity.

In case of element-wise constant elevation/layer position:

- spatial derivations require correction terms due to discontinuities;
- Leibniz's formula does not hold (i.e. requires additional correction terms) if layers are not face to face from one column to another;
- *most known models do not care properly about those two issues, may it be because it would significantly increase computation time?*

3D σ-layers, finite volumes formulation

Finite differences equations (change of vertical coordinates) can be seen as layer-integrated equations, with application of Leibniz's formula

Volume conservation: $0 = \int_{s_0}^{s_1} \nabla \cdot \boldsymbol{u} ds = \int_{s_0}^{s_1} \nabla_H \cdot \boldsymbol{v} dz + [w]_{s_0}^{s_1}$

According to Leibniz's formula: $\int_{s_0}^{s_1} \nabla \cdot \boldsymbol{u} ds = \nabla_H \cdot \int_{s_0}^{s_1} \boldsymbol{v} dz - [\boldsymbol{v} \cdot \nabla_H s]$

$$\text{Then: } \int_{s_0}^{s_1} \nabla \cdot \boldsymbol{u} ds = \nabla_H \cdot \int_{s_0}^{s_1} \boldsymbol{v} dz + [\omega]_{s_0}^{s_1} + \left[\frac{\partial s}{\partial t}\right]_{s_0}^{s_1} \quad (16)$$

$\boldsymbol{u}$: 3D velocity;
$\boldsymbol{v}$: horizontal velocity;
w: vertical velocity;
ω: omega velocity.

In case of element-wise constant elevation/layer position:

- spatial derivations require correction terms due to discontinuities;
- Leibniz's formula does not hold (i.e. requires additional correction terms) if layers are not face to face from one column to another;
- *most known models do not care properly about those two issues, may it be because it would significantly increase computation time?*

3D σ-layers, finite volumes formulation

Finite differences equations (change of vertical coordinates) can be seen as layer-integrated equations, with application of Leibniz's formula

Volume conservation: $0 = \int_{s_0}^{s_1} \nabla \cdot \boldsymbol{u} ds = \int_{s_0}^{s_1} \nabla_H \cdot \boldsymbol{v} dz + [w]_{s_0}^{s_1}$

According to Leibniz's formula: $\int_{s_0}^{s_1} \nabla \cdot \boldsymbol{u} ds = \nabla_H \cdot \int_{s_0}^{s_1} \boldsymbol{v} dz - [\boldsymbol{v} \cdot \nabla_H s]$

Then:
$$\int_{s_0}^{s_1} \nabla \cdot \boldsymbol{u} ds = \nabla_H \cdot \int_{s_0}^{s_1} \boldsymbol{v} dz + [\omega]_{s_0}^{s_1} + \left[\frac{\partial s}{\partial t}\right]_{s_0}^{s_1} \qquad (16)$$

$\boldsymbol{u}$: 3D velocity;
$\boldsymbol{v}$: horizontal velocity;
w: vertical velocity;
ω: omega velocity.

In case of element-wise constant elevation/layer position:

- spatial derivations require correction terms due to discontinuities;
- Leibniz's formula does not hold (i.e. requires additional correction terms) if layers are not face to face from one column to another;
- *most known models do not care properly about those two issues, may it be because it would significantly increase computation time?*

3D σ-layers, finite volumes formulation

Finite differences equations (change of vertical coordinates) can be seen as layer-integrated equations, with application of Leibniz's formula

Volume conservation: $0 = \int_{s_0}^{s_1} \nabla \cdot \boldsymbol{u} ds = \int_{s_0}^{s_1} \nabla_H \cdot \boldsymbol{v} dz + [w]_{s_0}^{s_1}$

According to Leibniz's formula: $\int_{s_0}^{s_1} \nabla \cdot \boldsymbol{u} ds = \nabla_H \cdot \int_{s_0}^{s_1} \boldsymbol{v} dz - [\boldsymbol{v} \cdot \nabla_H s]$

$$\text{Then: } \int_{s_0}^{s_1} \nabla \cdot \boldsymbol{u} ds = \nabla_H \cdot \int_{s_0}^{s_1} \boldsymbol{v} dz + [\omega]_{s_0}^{s_1} + \left[\frac{\partial s}{\partial t}\right]_{s_0}^{s_1} \quad (16)$$

$\boldsymbol{u}$: 3D velocity;
$\boldsymbol{v}$: horizontal velocity;
w: vertical velocity;
ω: omega velocity.

In case of element-wise constant elevation/layer position:

- spatial derivations require correction terms due to discontinuities;
- Leibniz's formula does not hold (i.e. requires additional correction terms) if layers are not face to face from one column to another;
- *most known models do not care properly about those two issues, may it be because it would significantly increase computation time?*

3D σ-layers, finite volumes formulation

Finite differences equations (change of vertical coordinates) can be seen as layer-integrated equations, with application of Leibniz's formula

Volume conservation: $0 = \int_{s_0}^{s_1} \boldsymbol{\nabla} \cdot \boldsymbol{u} ds = \int_{s_0}^{s_1} \boldsymbol{\nabla}_H \cdot \boldsymbol{v} dz + [w]_{s_0}^{s_1}$

According to Leibniz's formula: $\int_{s_0}^{s_1} \boldsymbol{\nabla} \cdot \boldsymbol{u} ds = \boldsymbol{\nabla}_H \cdot \int_{s_0}^{s_1} \boldsymbol{v} dz - [\boldsymbol{v} \cdot \boldsymbol{\nabla}_H s]$

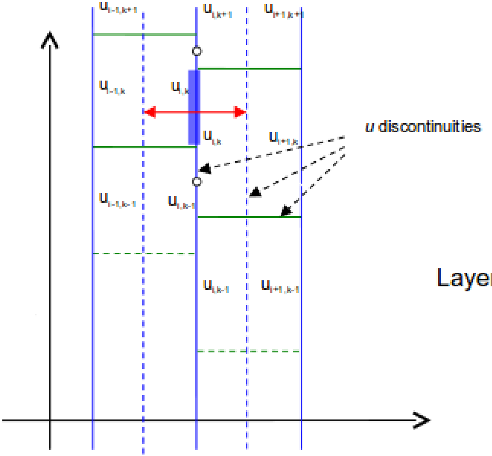
$$\text{Then: } \int_{s_0}^{s_1} \boldsymbol{\nabla} \cdot \boldsymbol{u} ds = \boldsymbol{\nabla}_H \cdot \int_{s_0}^{s_1} \boldsymbol{v} dz + [\omega]_{s_0}^{s_1} + \left[\frac{\partial s}{\partial t}\right]_{s_0}^{s_1} \quad (16)$$

$\boldsymbol{u}$: 3D velocity;
$\boldsymbol{v}$: horizontal velocity;
w: vertical velocity;
ω: omega velocity.

In case of element-wise constant elevation/layer position:

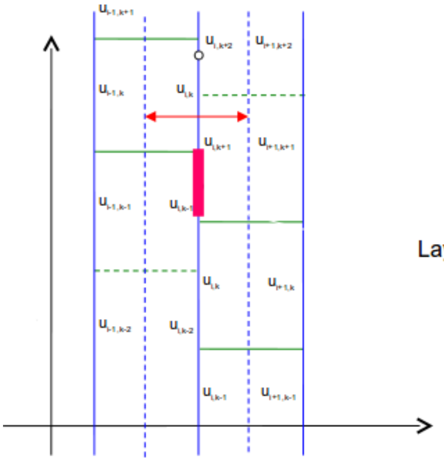
- spatial derivations require correction terms due to discontinuities;
- Leibniz's formula does not hold (i.e. requires additional correction terms) if layers are not face to face from one column to another;
- *most known models do not care properly about those two issues, may it be because it would significantly increase computation time?*

Leibniz's formula holds in this case



Layers have a partial continuity

Leibniz's formula does not hold in this case



Layers are disjoint

Some partial conclusions 1/2

- Most of FD, FV and FE dynamical and tracers discrete equations are identical if discontinuities are properly treated;
- laplacian operator (for piece-wise constant fields) needs further investigation:
  - simple extension from 1D to 2D does not work in general;
  - FD and FE may differ significantly;
  - explicit, well controlled smoothing (such as forward-backward projection on a different discretisation) might be an way-through (there is already a lot of hidden smoothing in most models);
  - two approaches to determine discrete laplacian, can we settle them (measure)?

Some partial conclusions 1/2

- Most of FD, FV and FE dynamical and tracers discrete equations are identical if discontinuities are properly treated;
- laplacian operator (for piece-wise constant fields) needs further investigation:
  - simple extension from 1D to 2D does not work in general;
  - FD and FE may differ significantly;
  - explicit, well controlled smoothing (such as forward-backward projection on a different discretisation) might be an way-through (there is already a lot of hidden smoothing in most models);
  - two approaches to determine discrete laplacian, can we settle them (measure)?

Some partial conclusions 1/2

- Most of FD, FV and FE dynamical and tracers discrete equations are identical if discontinuities are properly treated;
- laplacian operator (for piece-wise constant fields) needs further investigation:
  - simple extension from 1D to 2D does not work in general;
  - FD and FE may differ significantly;
  - explicit, well controlled smoothing (such as forward-backward projection on a different discretisation) might be an way-through (there is already a lot of hidden smoothing in most models);
  - two approaches to determine discrete laplacian, can we settle them (measure)?

Some partial conclusions 1/2

- Most of FD, FV and FE dynamical and tracers discrete equations are identical if discontinuities are properly treated;
- laplacian operator (for piece-wise constant fields) needs further investigation:
  - simple extension from 1D to 2D does not work in general;
  - FD and FE may differ significantly;
  - explicit, well controlled smoothing (such as forward-backward projection on a different discretisation) might be an way-through (there is already a lot of hidden smoothing in most models);
  - two approaches to determine discrete laplacian, can we settle them (measure)?

Some partial conclusions 1/2

- Most of FD, FV and FE dynamical and tracers discrete equations are identical if discontinuities are properly treated;
- laplacian operator (for piece-wise constant fields) needs further investigation:
  - simple extension from 1D to 2D does not work in general;
  - FD and FE may differ significantly;
  - explicit, well controlled smoothing (such as forward-backward projection on a different discretisation) might be an way-through (there is already a lot of hidden smoothing in most models);
  - two approaches to determine discrete laplacian, can we settle them (measure)?

Some partial conclusions 1/2

- Most of FD, FV and FE dynamical and tracers discrete equations are identical if discontinuities are properly treated;
- laplacian operator (for piece-wise constant fields) needs further investigation:
  - simple extension from 1D to 2D does not work in general;
  - FD and FE may differ significantly;
  - explicit, well controlled smoothing (such as forward-backward projection on a different discretisation) might be an way-through (there is already a lot of hidden smoothing in most models);
  - two approaches to determine discrete laplacian, can we settle them (measure)?

Some partial conclusions 2/2

- Discontinuous elevation/layer positions are a real issue:
  - if treated properly, suppress computational interest of FV;
  - if not treated properly, lead to inconsistencies (such as the hydrostatic one). In most FV formulation, elevation is piece-wise constant in mass conservation computation, continuous in pressure gradient computation (thus inconsistent);
- structured and unstructured models suffer the same issue, except that variable resolution can help in large bathymetry gradient regions (such as the continental shelf slopes);
- we have started to investigate a new 3D discretisation that allows to keep FV schemes with continuous elevation/layers (to be continued).

Some partial conclusions 2/2

- Discontinuous elevation/layer positions are a real issue:
  - if treated properly, suppress computational interest of FV;
  - if not treated properly, lead to inconsistencies (such as the hydrostatic one). In most FV formulation, elevation is piece-wise constant in mass conservation computation, continuous in pressure gradient computation (thus inconsistent);
- structured and unstructured models suffer the same issue, except that variable resolution can help in large bathymetry gradient regions (such as the continental shelf slopes);
- we have started to investigate a new 3D discretisation that allows to keep FV schemes with continuous elevation/layers (to be continued).

Some partial conclusions 2/2

- Discontinuous elevation/layer positions are a real issue:
  - if treated properly, suppress computational interest of FV;
  - if not treated properly, lead to inconsistencies (such as the hydrostatic one). In most FV formulation, elevation is piece-wise constant in mass conservation computation, continuous in pressure gradient computation (thus inconsistent);
- structured and unstructured models suffer the same issue, except that variable resolution can help in large bathymetry gradient regions (such as the continental shelf slopes);
- we have started to investigate a new 3D discretisation that allows to keep FV schemes with continuous elevation/layers (to be continued).

Some partial conclusions 2/2

- Discontinuous elevation/layer positions are a real issue:
  - if treated properly, suppress computational interest of FV;
  - if not treated properly, lead to inconsistencies (such as the hydrostatic one). In most FV formulation, elevation is piece-wise constant in mass conservation computation, continuous in pressure gradient computation (thus inconsistent);
- structured and unstructured models suffer the same issue, except that variable resolution can help in large bathymetry gradient regions (such as the continental shelf slopes);
- we have started to investigate a new 3D discretisation that allows to keep FV schemes with continuous elevation/layers (to be continued).

Some partial conclusions 2/2

- Discontinuous elevation/layer positions are a real issue:
  - if treated properly, suppress computational interest of FV;
  - if not treated properly, lead to inconsistencies (such as the hydrostatic one). In most FV formulation, elevation is piece-wise constant in mass conservation computation, continuous in pressure gradient computation (thus inconsistent);
- structured and unstructured models suffer the same issue, except that variable resolution can help in large bathymetry gradient regions (such as the continental shelf slopes);
- we have started to investigate a new 3D discretisation that allows to keep FV schemes with continuous elevation/layers (to be continued).

References

Le Bars, Yoann (2010). “Modélisation de la dynamique océanique barotrope dans l’estuaire et le plateau amazoniens”. French. PhD thesis. 118, route de Narbonne – F-31062 Toulouse cedex 9, France: université Toulouse 3 – Paul Sabatier.

Lyard, Florent and Yoann Le Bars (in prep.). “Gradient, divergence and laplacian approximation for numerical ocean modelling”. English.

Thank you