Spectral Lagrangian methods (in particular for SW)

L. Micalizzi[1,2], S. Tokareva[2], M. Ricchiuto[3], R. Abgrall[1]

[1]Institut für Mathematik,
Universität Zürich

[2]Theoretical Division,
Los Alamos National Laboratory

[3]Team CARDAMOM,
Inria Bordeaux sud-ouest

Outline

- Governing equations
- Scheme and motivation
- Numerics
- Continuous FEM
- Well-balancing
  - Spatial part
  - Stabilization, continuous interior penalty
- Numerical results

Section 1

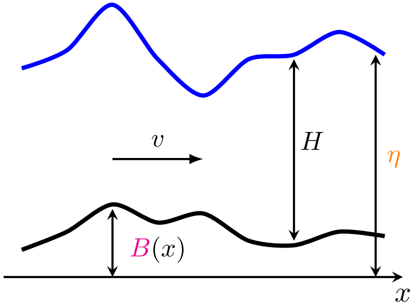
Governing equations

$$\frac{\partial}{\partial t}\boldsymbol{u} + div_{\boldsymbol{x}}\boldsymbol{F}(\boldsymbol{u}) = \boldsymbol{S}(\boldsymbol{x},\boldsymbol{u}), \quad (\boldsymbol{x},t) \in \Omega \times [0,T]$$

$$\boldsymbol{u} := \begin{pmatrix} H \\ H\boldsymbol{v} \end{pmatrix}$$

$$\boldsymbol{F}(\boldsymbol{u}) := \begin{pmatrix} H\boldsymbol{v} \\ H\boldsymbol{v}\otimes\boldsymbol{v} + g\frac{H^2}{2}\mathbb{I} \end{pmatrix}$$

$$\boldsymbol{S}(\boldsymbol{x},\boldsymbol{u}) := -\begin{pmatrix} 0 \\ gH\nabla_{\boldsymbol{x}}B(x) \end{pmatrix}$$



$$\eta := H + B$$

Shallow water equations, simplification of Euler (with gravity)

$$\frac{\partial}{\partial t}\boldsymbol{u} + div_{\boldsymbol{x}}\boldsymbol{F}(\boldsymbol{u}) = \boldsymbol{S}(\boldsymbol{x},\boldsymbol{u}), \quad (\boldsymbol{x},t) \in \Omega \times [0,T]$$

$$\boldsymbol{u} := \begin{pmatrix} H \\ H\boldsymbol{v} \end{pmatrix}$$

$$\boldsymbol{F}(\boldsymbol{u}) := \begin{pmatrix} H\boldsymbol{v} \\ H\boldsymbol{v}\otimes\boldsymbol{v} + g\frac{H^2}{2}\mathbb{I} \end{pmatrix}$$

$$\boldsymbol{S}(\boldsymbol{x},\boldsymbol{u}) := -\begin{pmatrix} 0 \\ gH\nabla_{\boldsymbol{x}}B(x) \end{pmatrix}$$

$$H \sim \rho$$

$$p \sim g\frac{H^2}{2}$$

$$B(x) \sim \phi(x)$$

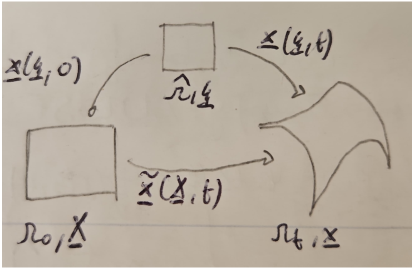
Neglecting the energy equation

Section 2

Scheme and motivation

Shallow water equations, Lagrangian

$$\Omega_0 \ni \boldsymbol{X} \longrightarrow \boldsymbol{x} = \widetilde{\boldsymbol{x}}(\boldsymbol{X}, t) \in \Omega_t$$
$$\widehat{\Omega} \ni \boldsymbol{\xi} \longrightarrow \boldsymbol{x} = \boldsymbol{x}(\boldsymbol{\xi}, t) \in \Omega_t$$
$$\boldsymbol{X} = \boldsymbol{x}(\boldsymbol{\xi}, 0)$$



Shallow water equations, Lagrangian

- **Motion**

$$\frac{d}{dt}\boldsymbol{x}(\boldsymbol{\xi},t) = \boldsymbol{v}(\boldsymbol{x}(\boldsymbol{\xi},t),t)$$

- **Water height/density (SMC)**

$$H(\boldsymbol{x}(\boldsymbol{\xi},t),t) = \frac{\widehat{H}(\boldsymbol{\xi})}{det\boldsymbol{J}(\boldsymbol{\xi},t)}, \quad \boldsymbol{J}(\boldsymbol{\xi},t) = \frac{\partial \boldsymbol{x}}{\partial \boldsymbol{\xi}}(\boldsymbol{\xi},t)$$

- **Velocity**

$$\begin{aligned} H(\boldsymbol{x},t)\frac{d}{dt}\boldsymbol{v}(\boldsymbol{x},t) &= -\nabla_{\boldsymbol{x}} p - gH\nabla_{\boldsymbol{x}} B \\ &= -\nabla_{\boldsymbol{x}}\left(\frac{gH^2}{2}\right) - gH\nabla_{\boldsymbol{x}} B \end{aligned}$$

$$\frac{d}{dt}\boldsymbol{v}(\boldsymbol{x},t) = -g\nabla_{\boldsymbol{x}}(H+B)$$

$\widehat{\varphi}_i \in \mathbb{P}_{M+1}$ continuous
$\widehat{\psi}_i \in \mathbb{P}_M$ discontinuous

$$\varphi_i(\boldsymbol{x}(\boldsymbol{\xi},t),t) = \widehat{\varphi}_i(\boldsymbol{\xi})$$
$$\psi_i(\boldsymbol{x}(\boldsymbol{\xi},t),t) = \widehat{\psi}_i(\boldsymbol{\xi})$$

$$\boldsymbol{x}_h(\boldsymbol{\xi},t) = \sum_i \boldsymbol{x}_i(t)\widehat{\varphi}_i(\boldsymbol{\xi})$$
$$\boldsymbol{v}_h(\boldsymbol{\xi},t) = \sum_i \boldsymbol{v}_i(t)\widehat{\varphi}_i(\boldsymbol{\xi})$$
$$H_h(\boldsymbol{\xi},t) = \sum_i H_i(t)\widehat{\psi}_i(\boldsymbol{\xi})$$

$$\boldsymbol{v}_h(\boldsymbol{x},t) = \sum_i \boldsymbol{v}_i(t)\widehat{\varphi}_i(\boldsymbol{\xi}(\boldsymbol{x},t))$$
$$H_h(\boldsymbol{x},t) = \sum_i H_i(t)\widehat{\psi}_i(\boldsymbol{\xi}(\boldsymbol{x},t))$$

Shallow water equations, discretized equations

- **Motion**

$$\frac{d}{dt}\boldsymbol{x}_i(t) = \boldsymbol{v}_i(t)$$

- **Water height/density (SMC) $\Rightarrow$ PP**

$$H_i(t) = \frac{\widehat{H}_i}{det\boldsymbol{J}(\boldsymbol{\xi}_i, t)}, \quad \boldsymbol{J}(\boldsymbol{\xi}, t) = \frac{\partial \boldsymbol{x}}{\partial \boldsymbol{\xi}}(\boldsymbol{\xi}, t)$$

- **Velocity**

$$\sum_{K \in K_i} \sum_{\boldsymbol{x}_j \in K} \left(\int_K \varphi_i \varphi_j d\boldsymbol{x} \right) \frac{d}{dt}\boldsymbol{v}_j(t)$$

$$= -g \sum_{K \in K_i} \left[\int_K \varphi_i \nabla_{\boldsymbol{x}} (H_h + B_h) d\boldsymbol{x} \right] - \sum_{K \in K_i} \boldsymbol{CT}_i^K - \boldsymbol{ST}_i$$

Shallow water equations, discretized equations

$$\sum_{K\in K_i}\sum_{\boldsymbol{x}_j\in K}\left(\int_K \varphi_i\varphi_j d\boldsymbol{x}\right)\frac{d}{dt}\boldsymbol{v}_j(t)$$
$$= -g\sum_{K\in K_i}\left[\int_K \varphi_i \nabla_{\boldsymbol{x}}(H_h+B_h)d\boldsymbol{x}\right] - \sum_{K\in K_i}\boldsymbol{CT}_i^K - \boldsymbol{ST}_i$$

$$\boldsymbol{CT}_i^K = g\int_{\partial K}\varphi_i(\eta^* - \eta|_K)\boldsymbol{\nu}d\boldsymbol{\sigma}$$

$$\boldsymbol{ST}_i^{CIP} := \sum_f \alpha_f \left[\!\left[\frac{\partial}{\partial x}\varphi_i\right]\!\right]\left[\!\left[\frac{\partial}{\partial x}v\right]\!\right], \quad \boldsymbol{ST}_i^{LxF} := \sum_{K\in K_i}\alpha_K(\boldsymbol{v}_i - \overline{\boldsymbol{v}}_K)$$

Last but not least an ODE integrator

Problem: mass matrix

$$\sum_{K\in K_i}\sum_{\boldsymbol{x}_j\in K}\left(\int_K \varphi_i\varphi_j d\boldsymbol{x}\right)\frac{d}{dt}\boldsymbol{v}_j(t)$$
$$= -g\sum_{K\in K_i}\left[\int_K \varphi_i \nabla_{\boldsymbol{x}}(H_h+B_h)d\boldsymbol{x}\right] - \sum_{K\in K_i}\boldsymbol{CT}_i^K - \boldsymbol{ST}_i$$

$$\mathcal{M}\frac{d}{dt}\boldsymbol{v} = \boldsymbol{r}$$

Solutions

- LO mass lumping
- DeC Remi[1] (problems for order 4 on)
- Spectral methods $\varphi_i \sim \boldsymbol{x}_i$ (1D GLB, 2D tensor product or Cubature)

[1] Abgrall, High order schemes for hyperbolic problems using globally continuous approximation and avoiding mass matrices, 2017

Price to pay: time-dependent mass matrix

$$\int_{K(t)} \varphi_i(\boldsymbol{x},t)\varphi_j(\boldsymbol{x},t)d\boldsymbol{x} = \int_{\widehat{K}} \widehat{\varphi}_i(\boldsymbol{\xi})\widehat{\varphi}_j(\boldsymbol{\xi})det\boldsymbol{J}(\boldsymbol{\xi},t)d\boldsymbol{\xi}$$
$$\approx \delta_{i,j}\widehat{\omega}_i det\boldsymbol{J}(\boldsymbol{\xi}_i,t)$$

$$\int_{K(t)} H_h(\boldsymbol{x},t)\varphi_i(\boldsymbol{x},t)\varphi_j(\boldsymbol{x},t)d\boldsymbol{x} = \int_{\widehat{K}} H_h(\boldsymbol{\xi},t)\widehat{\varphi}_i(\boldsymbol{\xi})\widehat{\varphi}_j(\boldsymbol{\xi})det\boldsymbol{J}(\boldsymbol{\xi},t)d\boldsymbol{\xi}$$
$$\approx \delta_{i,j}\widehat{\omega}_i H_h(\boldsymbol{\xi}_i,t)det\boldsymbol{J}(\boldsymbol{\xi}_i,t)$$

However, only a point evaluation

Advantages

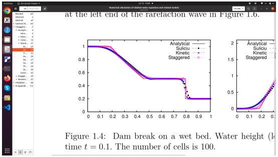
$$\frac{d}{dt}\boldsymbol{x} = \boldsymbol{v}, \quad \frac{d}{dt}\boldsymbol{v} = \boldsymbol{r}, \quad \text{SMC}$$

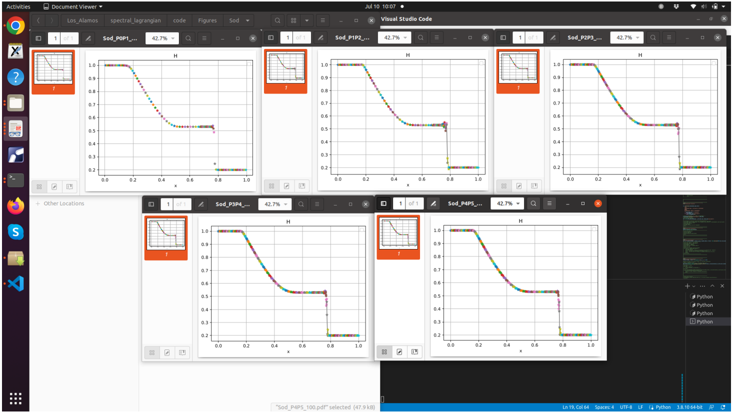
- Truly arbitrary high order (no problems as for Bernstein and β-limiting)
- Extendible to Euler
- Extendible to multi-D (quads, PGL, tensor products, Cubature)
- Whatever time integration method.

Section 3

Numerics

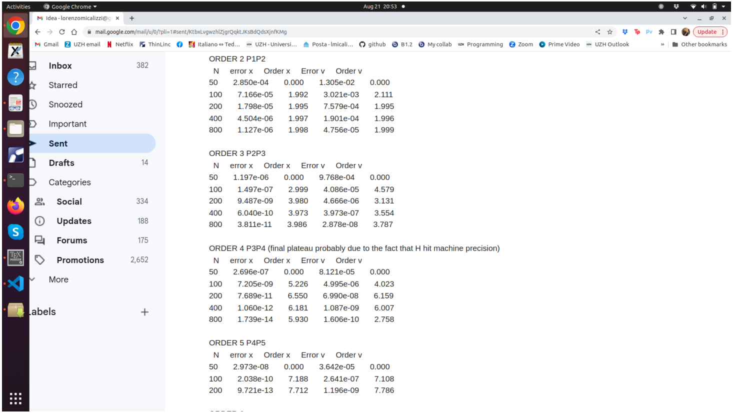
Numerics, Sod (Sorry for low quality)





Numerics, Convergence[2] ($2p$)

$$\begin{cases} H = 2 + cos(2\pi x) \\ v = 1 \end{cases} \tag{1}$$



ORDER 2 P1P2

N	error x	Order x	Error v	Order v
50	2.850e-04	0.000	1.305e-02	0.000
100	7.166e-05	1.992	3.021e-03	2.111
200	1.798e-05	1.995	7.579e-04	1.995
400	4.504e-06	1.997	1.901e-04	1.996
800	1.127e-06	1.998	4.756e-05	1.999

ORDER 3 P2P3

N	error x	Order x	Error v	Order v
50	1.197e-06	0.000	9.768e-04	0.000
100	1.497e-07	2.999	4.086e-05	4.579
200	9.487e-09	3.980	4.666e-06	3.131
400	6.040e-10	3.973	3.973e-07	3.554
800	3.811e-11	3.986	2.878e-08	3.787

ORDER 4 P3P4 (final plateau probably due to the fact that H hit machine precision)

N	error x	Order x	Error v	Order v
50	2.696e-07	0.000	8.121e-05	0.000
100	7.205e-09	5.226	4.995e-06	4.023
200	7.689e-11	6.550	6.990e-08	6.159
400	1.060e-12	6.181	1.087e-09	6.007
800	1.739e-14	5.930	1.606e-10	2.758

ORDER 5 P4P5

N	error x	Order x	Error v	Order v
50	2.973e-08	0.000	3.642e-05	0.000
100	2.038e-10	7.188	2.641e-07	7.108
200	9.721e-13	7.712	1.196e-09	7.786

[2] Mantri, Oeffner, Ricchiuto, Fully well balanced entropy controlled DGSEM, 2022

Numerics, well-balancing

In particular (in Eulerian) $\frac{\partial}{\partial t}\boldsymbol{u} \equiv 0 \;\Leftrightarrow\; \frac{\partial}{\partial x}\boldsymbol{F}(\boldsymbol{u}) = \boldsymbol{S}(x,\boldsymbol{u})$)

- Refine a lot the mesh $\Rightarrow$ Longer computational time
- Well-balancing

$$\boldsymbol{u}_{eq} := \begin{pmatrix} \eta - B \\ 0 \end{pmatrix}$$

$$\eta = H + B \equiv const$$

Figure: Lake of Zürich at rest

Non-smooth bathymetry, only C^0

$$\eta(x) = \begin{cases} \eta_{eq} + A \exp\left(1 - \frac{1}{1-\left(\frac{x-6}{0.5}\right)^2}\right) & 5.5 < x < 6.5 \\ \eta_{eq} & otherwise \end{cases}$$

with $A = 5 \cdot 10^{-5}$.

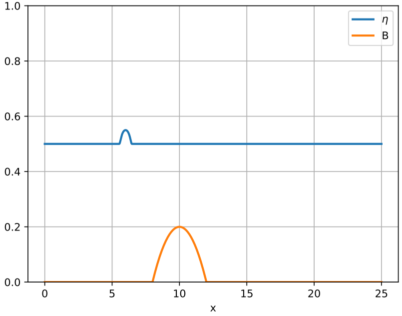
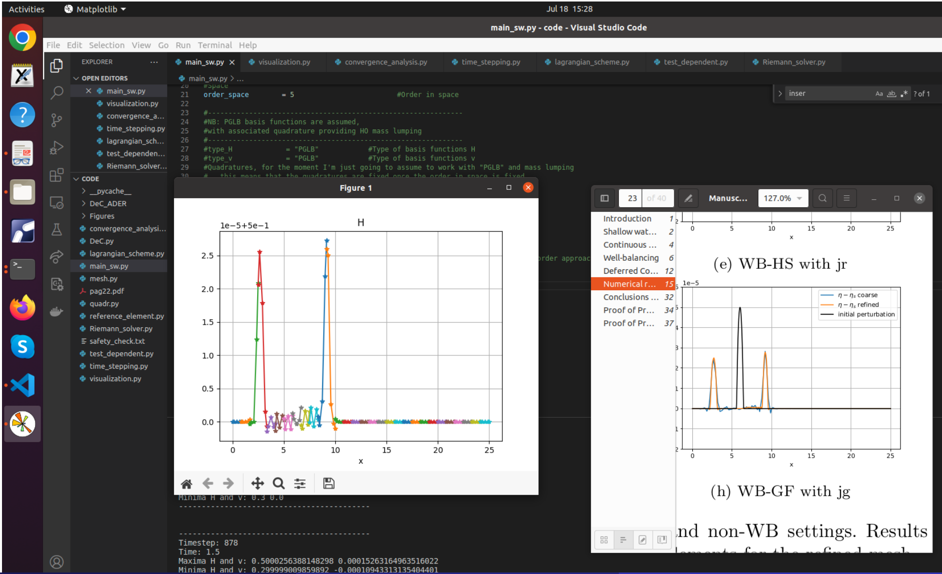


Figure: Perturbation amplified by 1000

Numerics, well-balancing



Section 4

Continuous FEM

Continuous FEM

- A tessellation $\mathcal{T}_h = \{K\}$ of Ω;

Continuous FEM

- A tessellation $\mathcal{T}_h = \{K\}$ of Ω;
- The space V_h of continuous piecewise polynomial functions;

Continuous FEM

- A tessellation $\mathcal{T}_h = \{K\}$ of Ω;
- The space V_h of continuous piecewise polynomial functions;
- a basis $\{\varphi_i\}_{i=1,\dots,I}$ of V_h (Bn, Pn, PGLn);

Continuous FEM

- A tessellation $\mathcal{T}_h = \{K\}$ of Ω;
- The space V_h of continuous piecewise polynomial functions;
- a basis $\{\varphi_i\}_{i=1,\dots,I}$ of V_h (Bn, Pn, PGLn);
- We look for $\boldsymbol{u}_h(x,t) := \sum_{j=1}^{I} \boldsymbol{c}_j(t)\varphi_j(x)$ s.t.

$$\int_\Omega \left(\frac{\partial}{\partial t}\boldsymbol{u}_h + \left[\frac{\partial}{\partial x}\boldsymbol{F}(\boldsymbol{u}_h) - \boldsymbol{S}(x,\boldsymbol{u}_h) \right]_h \right) \varphi_i(x) + \mathsf{ST}_i(\boldsymbol{u}_h) = \boldsymbol{0} \quad \forall i$$

Continuous FEM

- A tessellation $\mathcal{T}_h = \{K\}$ of Ω;
- The space V_h of continuous piecewise polynomial functions;
- a basis $\{\varphi_i\}_{i=1,\dots,I}$ of V_h (Bn, Pn, PGLn);
- We look for $\boldsymbol{u}_h(x,t) := \sum_{j=1}^{I} \boldsymbol{c}_j(t)\varphi_j(x)$ s.t.

$$\int_\Omega \left(\frac{\partial}{\partial t}\boldsymbol{u}_h + \left[\frac{\partial}{\partial x}\boldsymbol{F}(\boldsymbol{u}_h) - \boldsymbol{S}(x,\boldsymbol{u}_h) \right]_h \right) \varphi_i(x) + \mathbf{ST}_i(\boldsymbol{u}_h) = \mathbf{0} \quad \forall i$$

$$\Downarrow$$

$$\mathcal{M}\frac{d}{dt}\boldsymbol{c}(t) = \boldsymbol{H}(\boldsymbol{c}(t))$$

Continuous FEM

- A tessellation $\mathcal{T}_h = \{K\}$ of Ω;
- The space V_h of continuous piecewise polynomial functions;
- a basis $\{\varphi_i\}_{i=1,\dots,I}$ of V_h (Bn, Pn, PGLn);
- We look for $\boldsymbol{u}_h(x,t) := \sum_{j=1}^{I} \boldsymbol{c}_j(t)\varphi_j(x)$ s.t.

$$\int_\Omega \left(\frac{\partial}{\partial t}\boldsymbol{u}_h + \left[\frac{\partial}{\partial x}\boldsymbol{F}(\boldsymbol{u}_h) - \boldsymbol{S}(x,\boldsymbol{u}_h) \right]_h \right)\varphi_i(x) + \mathbf{ST}_i(\boldsymbol{u}_h) = \mathbf{0} \quad \forall i$$

$$\Downarrow$$

$$\mathcal{M}\frac{d}{dt}\boldsymbol{c}(t) = \boldsymbol{H}(\boldsymbol{c}(t))$$

$$\boldsymbol{u}_{eq} := \begin{pmatrix} \eta - B \\ 0 \end{pmatrix}, \quad \eta = H + B \equiv const$$

Section 5

Well-balancing

Subsection 1

Spatial part

Spatial part

$$\left[\frac{\partial}{\partial x}\boldsymbol{F}(\boldsymbol{u}_h)-\boldsymbol{S}(x,\boldsymbol{u}_h)\right]_h = \frac{\partial}{\partial x}\boldsymbol{F}_h(x) - \boldsymbol{S}_h(x)$$

$$\boldsymbol{F}_h(x) = \sum_j \boldsymbol{F}_j\varphi_j(x), \quad \boldsymbol{S}_h(x) = \sum_j \boldsymbol{S}_j\varphi_j(x)$$

Spatial part

$$\left[\frac{\partial}{\partial x}\boldsymbol{F}(\boldsymbol{u}_h)-\boldsymbol{S}(x,\boldsymbol{u}_h)\right]_h = \frac{\partial}{\partial x}\boldsymbol{F}_h(x) - \boldsymbol{S}_h(x) \neq 0 \text{ for lake at rest}$$

$$\boldsymbol{F}_h(x) = \sum_j \boldsymbol{F}_j\varphi_j(x), \quad \boldsymbol{S}_h(x) = \sum_j \boldsymbol{S}_j\varphi_j(x)$$

$$\left[\frac{\partial}{\partial x}\boldsymbol{F}(\boldsymbol{u}_h)-\boldsymbol{S}(x,\boldsymbol{u}_h)\right]_h = \frac{\partial}{\partial x}\boldsymbol{F}_h(x) - \boldsymbol{S}_h(x) \neq 0 \text{ for lake at rest}$$

$$\boldsymbol{F}_h(x) = \sum_j \boldsymbol{F}_j\varphi_j(x), \quad \boldsymbol{S}_h(x) = \sum_j \boldsymbol{S}_j\varphi_j(x)$$

Special WB discretization[3]

$$\left[\frac{\partial}{\partial x}\boldsymbol{F}(\boldsymbol{u}_h)-\boldsymbol{S}(x,\boldsymbol{u}_h)\right]_h = \left[\frac{\partial}{\partial x}\begin{pmatrix} Hv \\ Hv^2 + g\frac{H^2}{2}\end{pmatrix} + \begin{pmatrix} 0 \\ gH\frac{\partial}{\partial x}B\end{pmatrix}\right]_h$$

$$= \frac{\partial}{\partial x}\left[\begin{pmatrix} Hv \\ Hv^2\end{pmatrix}\right]_h + \begin{pmatrix} 0 \\ gH_h\frac{\partial}{\partial x}\left(H_h + B_h\right)\end{pmatrix}$$

[3] Ricchiuto and Bollermann. Stabilized residual distribution for shallow water simulations, 2009

Spatial part

$$\left[\frac{\partial}{\partial x}\boldsymbol{F}(\boldsymbol{u}_h) - \boldsymbol{S}(x,\boldsymbol{u}_h)\right]_h = \frac{\partial}{\partial x}\boldsymbol{F}_h(x) - \boldsymbol{S}_h(x) \neq 0 \text{ for lake at rest}$$

$$\boldsymbol{F}_h(x) = \sum_j \boldsymbol{F}_j \varphi_j(x), \quad \boldsymbol{S}_h(x) = \sum_j \boldsymbol{S}_j \varphi_j(x)$$

Special WB discretization[3]

$$\left[\frac{\partial}{\partial x}\boldsymbol{F}(\boldsymbol{u}_h) - \boldsymbol{S}(x,\boldsymbol{u}_h)\right]_h = \left[\frac{\partial}{\partial x}\begin{pmatrix} Hv \\ Hv^2 + g\frac{H^2}{2}\end{pmatrix} + \begin{pmatrix} 0 \\ gH\frac{\partial}{\partial x}B\end{pmatrix}\right]_h$$

$$= \frac{\partial}{\partial x}\left[\begin{pmatrix} Hv \\ Hv^2\end{pmatrix}\right]_h + \begin{pmatrix} 0 \\ gH_h\frac{\partial}{\partial x}\left(H_h + B_h\right)\end{pmatrix} = \boldsymbol{0}$$

[3] Ricchiuto and Bollermann. Stabilized residual distribution for shallow water simulations, 2009

Subsection 2

Stabilization, continuous interior penalty

Stabilization part

$$\int_\Omega \varphi_i(x)\frac{\partial}{\partial t}\boldsymbol{u}_h d\boldsymbol{x} + \int_\Omega \varphi_i(x)\left[\frac{\partial}{\partial x}\boldsymbol{F}(\boldsymbol{u}_h) - \boldsymbol{S}(x,\boldsymbol{u}_h)\right]_h + \mathsf{ST}_i(\boldsymbol{u}_h) = \mathbf{0} \quad \forall i$$

$$\int_\Omega \varphi_i(x)\frac{\partial}{\partial t}\boldsymbol{u}_h d\boldsymbol{x} + \int_\Omega \varphi_i(x)\left[\frac{\partial}{\partial x}\boldsymbol{F}(\boldsymbol{u}_h) - \boldsymbol{S}(x,\boldsymbol{u}_h)\right]_h + \mathsf{ST}_i(\boldsymbol{u}_h) = \boldsymbol{0} \quad \forall i$$

CIP[4] stabilization terms (**jump c**)

$$\begin{aligned}\mathsf{ST}_i(\boldsymbol{u}_h) &:= \sum_f \alpha_f \int_f [\![\nabla_{\boldsymbol{x}}\varphi_i]\!][\![\nabla_{\boldsymbol{x}}\boldsymbol{u}_h]\!]d\sigma \\ &= \sum_f \alpha_f \left[\!\!\left[\frac{\partial}{\partial x}\varphi_i\right]\!\!\right]\left[\!\!\left[\frac{\partial}{\partial x}\boldsymbol{u}_h\right]\!\!\right]\end{aligned}$$

[4] J. Douglas Jr and T. Dupont Interior Penalty Procedures for Elliptic and Parabolic Galerkin Methods, 1976

$$\int_\Omega \varphi_i(x)\frac{\partial}{\partial t}\boldsymbol{u}_h d\boldsymbol{x} + \int_\Omega \varphi_i(x)\left[\frac{\partial}{\partial x}\boldsymbol{F}(\boldsymbol{u}_h) - \boldsymbol{S}(x,\boldsymbol{u}_h)\right]_h + \mathsf{ST}_i(\boldsymbol{u}_h) = \mathbf{0} \quad \forall i$$

CIP[4] stabilization terms (**jump c**)

$$\begin{aligned}\mathsf{ST}_i(\boldsymbol{u}_h) &:= \sum_f \alpha_f \int_f [\![\nabla_{\boldsymbol{x}}\varphi_i]\!][\![\nabla_{\boldsymbol{x}}\boldsymbol{u}_h]\!]d\sigma \\ &= \sum_f \alpha_f \left[\!\!\left[\frac{\partial}{\partial x}\varphi_i\right]\!\!\right]\left[\!\!\left[\frac{\partial}{\partial x}\boldsymbol{u}_h\right]\!\!\right] \\ &= \sum_f \alpha_f \left[\!\!\left[\frac{\partial}{\partial x}\varphi_i\right]\!\!\right]\left[\!\!\left[\frac{\partial}{\partial x}\begin{pmatrix} H_h \\ (Hv)_h \end{pmatrix}\right]\!\!\right] \neq \mathbf{0}\end{aligned}$$

[4] J. Douglas Jr and T. Dupont Interior Penalty Procedures for Elliptic and Parabolic Galerkin Methods, 1976

- **jump t** (total height)

$$\mathsf{ST}_i(\boldsymbol{u}_h) := \sum_f \alpha_f \left[\!\!\left[\frac{\partial}{\partial x} \varphi_i \right]\!\!\right] \left[\!\!\left[\frac{\partial}{\partial x} \begin{pmatrix} \eta \\ Hv \end{pmatrix} \right]\!\!\right]$$

Stabilization part, novel well-balanced jumps

- **jump t** (total height)

$$\mathsf{ST}_i(\boldsymbol{u}_h) := \sum_f \alpha_f \left[\!\!\left[\frac{\partial}{\partial x}\varphi_i \right]\!\!\right] \left[\!\!\left[\frac{\partial}{\partial x} \begin{pmatrix} \eta \\ Hv \end{pmatrix} \right]\!\!\right]$$

- **jump e** (entropy variables)

$$\mathsf{ST}_i(\boldsymbol{u}_h) := \sum_f \alpha_f \left[\!\!\left[\frac{\partial}{\partial x}\varphi_i \right]\!\!\right] \frac{\partial \boldsymbol{u}}{\partial \boldsymbol{w}} \left[\!\!\left[\frac{\partial}{\partial x}\boldsymbol{w} \right]\!\!\right], \quad \boldsymbol{w} := \begin{pmatrix} g\eta - \frac{v^2}{2} \\ v \end{pmatrix}$$

Stabilization part, novel well-balanced jumps

- **jump t** (total height)

$$\mathsf{ST}_i(\boldsymbol{u}_h) := \sum_f \alpha_f \left[\!\left[\frac{\partial}{\partial x}\varphi_i \right]\!\right] \left[\!\left[\frac{\partial}{\partial x} \begin{pmatrix} \eta \\ Hv \end{pmatrix} \right]\!\right]$$

- **jump e** (entropy variables)

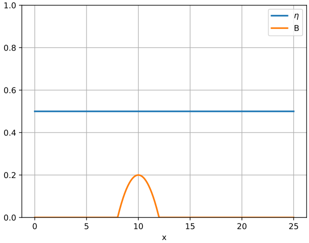
$$\mathsf{ST}_i(\boldsymbol{u}_h) := \sum_f \alpha_f \left[\!\left[\frac{\partial}{\partial x}\varphi_i \right]\!\right] \frac{\partial \boldsymbol{u}}{\partial \boldsymbol{w}} \left[\!\left[\frac{\partial}{\partial x}\boldsymbol{w} \right]\!\right], \quad \boldsymbol{w} := \begin{pmatrix} g\eta - \frac{v^2}{2} \\ v \end{pmatrix}$$

- **jump r** (residual)

$$\mathsf{ST}_i(\boldsymbol{u}_h) := \sum_f \alpha_f \left[\!\left[\boldsymbol{J}\frac{\partial}{\partial x}\varphi_i \right]\!\right] |\boldsymbol{J}|^{-1} \left[\!\left[\boldsymbol{J}\frac{\partial}{\partial x}\boldsymbol{u} - \boldsymbol{S} \right]\!\right], \quad \boldsymbol{J} := \frac{\partial \boldsymbol{F}}{\partial \boldsymbol{u}}$$

Section 6

Numerical results



Framework	L^1 error H	L^1 error Hv
WB-HS jc	3.007E-004	5.963E-004
WB-HS jt	9.403E-013	4.418E-012
WB-HS je	9.396E-013	4.415E-012
WB-HS jr	9.409E-013	4.415E-012

Table: PGL4 with 100 elements at $T_f = 10$

Same with Pn and Bn

Same domain, non-smooth bathymetry, only C^0

$$\eta(x) = \begin{cases} \eta_{eq} + A\exp\left(1 - \frac{1}{1-\left(\frac{x-6}{0.5}\right)^2}\right) & 5.5 < x < 6.5 \\ \eta_{eq} & otherwise \end{cases}$$

with $A = 5 \cdot 10^{-5}$.

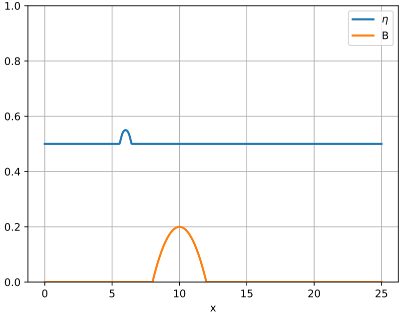
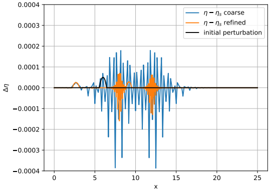


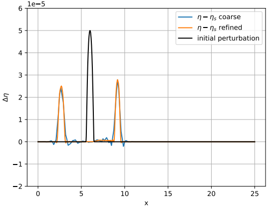
Figure: Perturbation amplified by 1000

Small perturbation of lake at rest

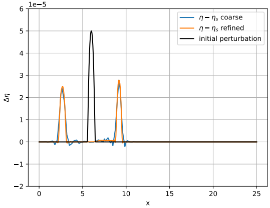
PGL4; coarse: 30 elements; refined: 128 elements



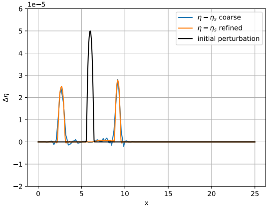
(a) Jump c



(b) Jump t



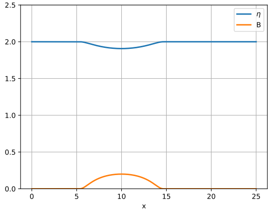
(c) Jump e



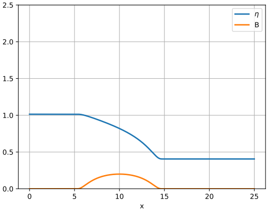
(d) Jump r

Question

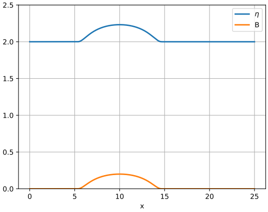
What about the other steady states?



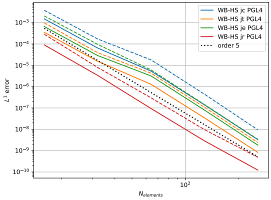
(a) Sub, $Hv \equiv 4.42$

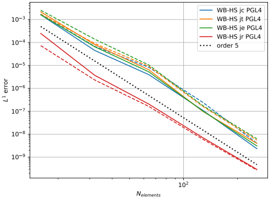


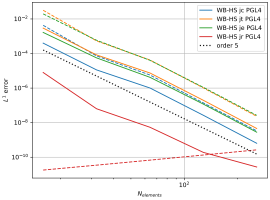
(b) Trans, $Hv \equiv 1.53$



(c) Super, $Hv \equiv 24$

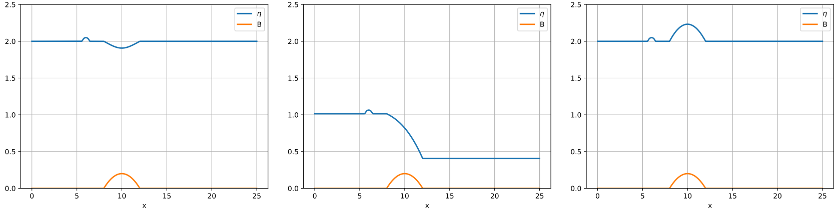






Continuous H, Dashed Hv

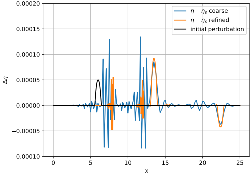
Small perturbation



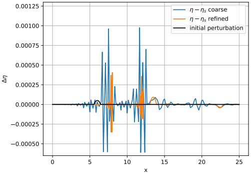
(a) Subcritical (b) Transcritical (c) Supercritical

Figure: Perturbation amplified by 1000

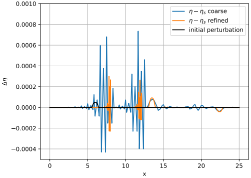
Supercritical



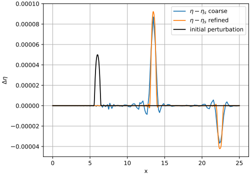
(a) Jump c



(b) Jump t

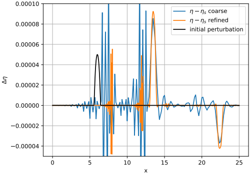


(c) Jump e

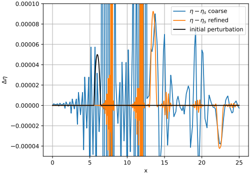


(d) Jump r

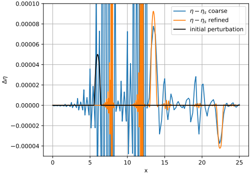
Supercritical



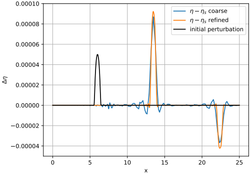
(a) Jump c



(b) Jump t



(c) Jump e



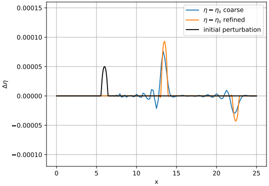
(d) Jump r

Analogous results

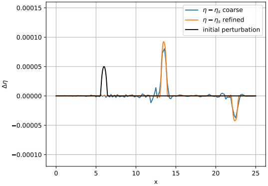
Analogous results for...

Analogous results

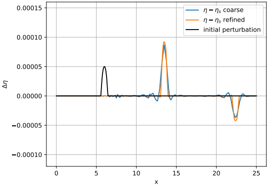
- for other basis functions (Bn, Pn and PGLn)



(a) PGL2



(b) PGL3

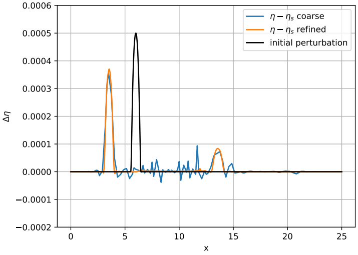


(c) PGL4

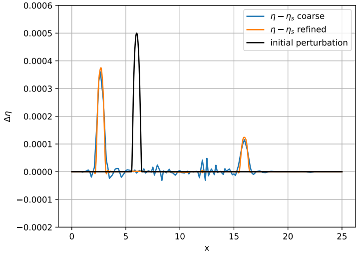
Supercritical: "fair" comparison with 60, 40 and 30 elements

(Constant number of DoFs)

- for other tests



(a) Transcritical

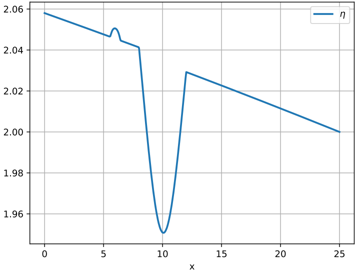


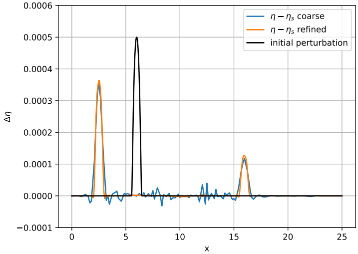
(b) Subcritical

(NB: No limiting at all)

- with friction

$$\boldsymbol{S}(x,\boldsymbol{u}) = -\begin{pmatrix} 0 \\ gH\frac{\partial}{\partial x}B \end{pmatrix} - g\frac{n^2|Hv|}{H^{\frac{7}{3}}}\begin{pmatrix} 0 \\ Hv \end{pmatrix}$$

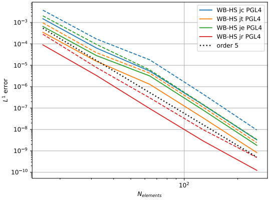




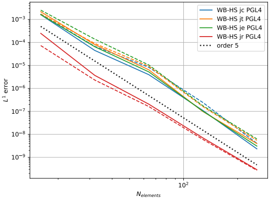
Subcritical: perturbation amplified by 100

(Same for supercritical)

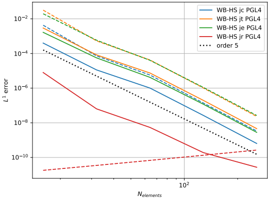
Global flux, $\boldsymbol{G} = \boldsymbol{F} - \int_{x_L}^{x} \boldsymbol{S}$



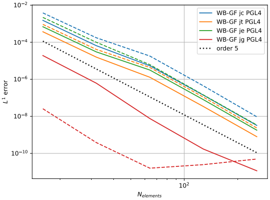
(a) Subcritical

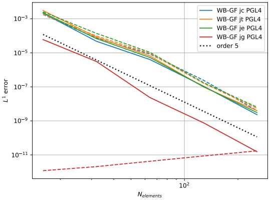


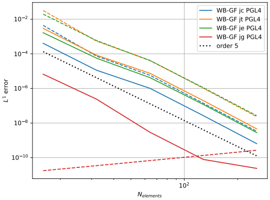
(b) Transcritical



(c) Supercritical







Section 7

Conclusions

Conclusions

Conclusions

- WB space discretization (w.r.t. lake at rest)

Conclusions

- WB space discretization (w.r.t. lake at rest)
- Three new well-balanced CIP stabilizations
  - jt (total height)
  - je (entropy variables)
  - jr (residual)

Conclusions

- WB space discretization (w.r.t. lake at rest)
- Three new well-balanced CIP stabilizations
  - jt (total height)
  - je (entropy variables)
  - jr (residual)
- The numerical results confirm
  - Exact well-balancing for lake at rest
  - HO accuracy
  - Ability of jr to handle general steady states

Thank you[5]

Novel well-balanced continuous interior penalty stabilizations;

Micalizzi, Ricchiuto, Abgrall; 2023

[5] Looking for a postdoc position in the U.S.

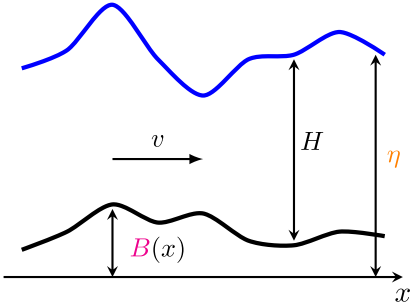
Shallow water equations, Eulerian

$$\frac{\partial}{\partial t}\boldsymbol{u} + \frac{\partial}{\partial x}\boldsymbol{F}(\boldsymbol{u}) = \boldsymbol{S}(x,\boldsymbol{u}), \quad (x,t) \in \Omega \times [0,T], \ \Omega := (x_L, x_R)$$

$$\boldsymbol{u} := \begin{pmatrix} H \\ Hv \end{pmatrix}$$

$$\boldsymbol{F}(\boldsymbol{u}) := \begin{pmatrix} Hv \\ Hv^2 + g\frac{H^2}{2} \end{pmatrix}$$

$$\boldsymbol{S}(x,\boldsymbol{u}) := -\begin{pmatrix} 0 \\ gH\frac{\partial}{\partial x}B(x) \end{pmatrix}$$



$$\eta := H + B$$

Shallow water equations, simplification of Euler (with gravity)

$$\frac{\partial}{\partial t}\boldsymbol{u} + \frac{\partial}{\partial x}\boldsymbol{F}(\boldsymbol{u}) = \boldsymbol{S}(x, \boldsymbol{u}), \quad (x,t) \in \Omega \times [0,T],\ \Omega := (x_L, x_R)$$

$$\boldsymbol{u} := \begin{pmatrix} H \\ Hv \end{pmatrix}$$

$$\boldsymbol{F}(\boldsymbol{u}) := \begin{pmatrix} Hv \\ Hv^2 + g\frac{H^2}{2} \end{pmatrix}$$

$$\boldsymbol{S}(x, \boldsymbol{u}) := -\begin{pmatrix} 0 \\ gH\frac{\partial}{\partial x}B(x) \end{pmatrix}$$

$$H \sim \rho$$

$$p \sim g\frac{H^2}{2}$$

$$B(x) \sim \phi(x)$$

Neglecting the energy equation