# Structure preserving methods via Global Flux quadrature: divergence-free preservation with continuous Finite Element

**Davide Torlo**, Mario Ricchiuto, Wasilij Barsukow

Dipartimento di Matematica "Guido Castelnuovo", Università di Roma La Sapienza, Italy
davidetorlo.it

Lisbon - 3rd June 2024

# SUPG

## SUPG

- Streamline **upwind** Petrov-Galerkin
- Stabilization for Continuous Galerkin Finite Element methods
- Advection dominated problems

# SUPG

## SUPG

- Streamline **upwind** Petrov-Galerkin
- Stabilization for Continuous Galerkin Finite Element methods
- Advection dominated problems

## SUPG for a **system** of linear hyperbolic conservation laws

$$\partial_t q + J^x \partial_x q + J^y \partial_y q = 0, \qquad q : \Omega_h \times \mathbb{R}^+ \to \mathbb{R}^S.$$

Take $V_h^K := \{\varphi \in \mathcal{C}(\Omega_h) : \varphi|_E \in \mathbb{P}^K(E) \forall E \in \Omega_h\}$. SUPG is $\forall \varphi \in (V_{h,0}^K)^S$ find $q \in (V_h^K)^S$ such that

$$0 = \int (\varphi) \left(\partial_t q + J^x \partial_x q + J^y \partial_y q\right) \mathrm{d}x$$

# SUPG

## SUPG

- Streamline **upwind** Petrov-Galerkin
- Stabilization for Continuous Galerkin Finite Element methods
- Advection dominated problems

## SUPG for a **system** of linear hyperbolic conservation laws

$$\partial_t q + J^x \partial_x q + J^y \partial_y q = 0, \qquad q : \Omega_h \times \mathbb{R}^+ \to \mathbb{R}^S.$$

Take $V_h^K := \{\varphi \in \mathcal{C}(\Omega_h) : \varphi|_E \in \mathbb{P}^K(E) \forall E \in \Omega_h\}$. SUPG is $\forall \varphi \in (V_{h,0}^K)^S$ find $q \in (V_h^K)^S$ such that

$$0 = \int \left(\varphi + \alpha \Delta x \partial_x \varphi \,\mathrm{sign}(J^x) + \alpha \Delta y \partial_y \varphi \,\mathrm{sign}(J^y)\right) \left(\partial_t q + J^x \partial_x q + J^y \partial_y q\right) \mathrm{d}x$$

# SUPG

## SUPG

- Streamline **upwind** Petrov-Galerkin
- Stabilization for Continuous Galerkin Finite Element methods
- Advection dominated problems

## SUPG for a **system** of linear hyperbolic conservation laws

$$\partial_t q + J^x \partial_x q + J^y \partial_y q = 0, \qquad q : \Omega_h \times \mathbb{R}^+ \to \mathbb{R}^S.$$

Take $V_h^K := \{\varphi \in \mathcal{C}(\Omega_h) : \varphi|_E \in \mathbb{P}^K(E) \forall E \in \Omega_h\}$. SUPG is $\forall \varphi \in (V_{h,0}^K)^S$ find $q \in (V_h^K)^S$ such that

$$\begin{aligned}
0 =& \int (\varphi + \alpha \Delta x \partial_x \varphi \,\mathrm{sign}(J^x) + \alpha \Delta y \partial_y \varphi \,\mathrm{sign}(J^y)) (\partial_t q + J^x \partial_x q + J^y \partial_y q) \,\mathrm{d}x \\
=& \int \varphi (\partial_t q + J^x \partial_x q + J^y \partial_y q) \mathrm{d}x + \alpha \int (\Delta x \partial_x \varphi \,\mathrm{sign}(J^x) + \Delta y \partial_y \varphi \,\mathrm{sign}(J^y)) \partial_t q \,\mathrm{d}x \\
&+ \alpha \int \Delta x \partial_x \varphi \,\mathrm{sign}(J^x)(J^x \partial_x q + J^y \partial_y q) \mathrm{d}x + \alpha \int \Delta y \partial_y \varphi \,\mathrm{sign}(J^y)(J^x \partial_x q + J^y \partial_y q) \mathrm{d}x.
\end{aligned}$$

# Acoustics equations and involutions

## Acoustics equation

$$\begin{cases} \partial_t u + \partial_x p = 0, \\ \partial_t v + \partial_y p = 0, \\ \partial_t p + \partial_x u + \partial_y v = 0, \end{cases}$$

$$\begin{cases} \partial_t \underline{v} + \nabla p = 0, \\ \partial_t p + \nabla \cdot \underline{v} = 0, \end{cases}$$

$$\partial_t q + J^x \partial_x q + J^y \partial_y q = 0$$

$$q = \begin{pmatrix} u \\ v \\ p \end{pmatrix}, \quad J^x = \begin{pmatrix} 0 & 0 & 1 \\ 0 & 0 & 0 \\ 1 & 0 & 0 \end{pmatrix}, \quad J^y = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \end{pmatrix}.$$

## Involution

The system of linear acoustics possesses an involution:

$$\partial_t (\nabla \times \underline{v}) = \nabla \times \partial_t \underline{v} = -\nabla \times \nabla p = 0,$$

## Equilibria

$$q : \partial_t q = 0, \qquad \begin{cases} \nabla \cdot \underline{v} = 0 \\ p \equiv C \in \mathbb{R} \end{cases}$$

## Other equations sharing div-free equilibria

SW, Euler, Maxwell, low Mach

# Typical problems

4.1 Low Mach number limit 81

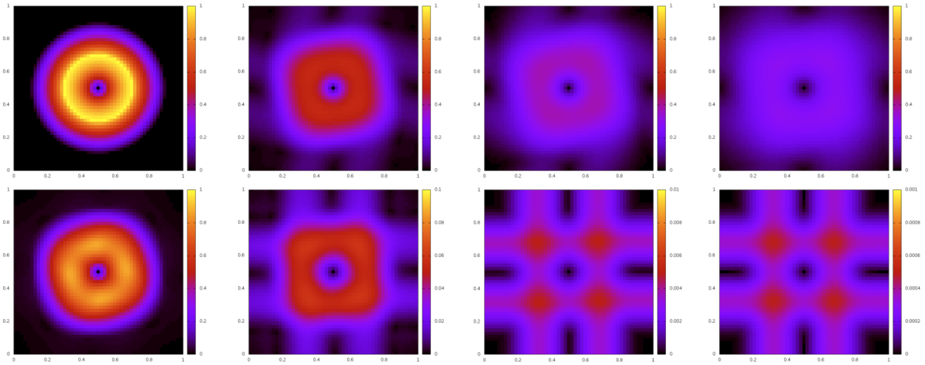


Figure 4.1: Simulation results for a vortex setup for $t = 0, 1, 2, 3$ (from left to right). Colour coded is $\sqrt{u^2 + v^2}$ *Top row*: Euler equations. *Bottom row*: Acoustic equations. [1,2]

[1] Barsukow, W. Low Mach number finite volume methods for the acoustic and Euler equations, Ph.D. thesis, 2018.
[2] Finite Volume Upwind numerical flux simulations.

# Typical problems: SUPG

Let's try with SUPG.

**Hope**

$$\int (\varphi + \alpha \Delta x \partial_x \varphi \,\text{sign}(J^x) + \alpha \Delta y \partial_y \varphi \,\text{sign}(J^y)) \begin{pmatrix} \partial_t u + \partial_x p \\ \partial_t v + \partial_y p \\ \partial_t p + \partial_x u + \partial_y v \end{pmatrix} \mathrm{d}x = 0$$

# Typical problems: SUPG

Let's try with SUPG.

## Hope

$$\int \left(\varphi + \alpha \Delta x \partial_x \varphi \,\mathrm{sign}(J^x) + \alpha \Delta y \partial_y \varphi \,\mathrm{sign}(J^y)\right) \begin{pmatrix} \partial_t u + \partial_x p \\ \partial_t v + \partial_y p \\ \partial_t p + \partial_x u + \partial_y v \end{pmatrix} \mathrm{d}x = 0$$

$$\int \varphi(\partial_t u + \partial_x p) + \alpha \Delta x \, \partial_x \varphi(\partial_t p + \partial_x u + \partial_y v) = 0$$

$$\int \varphi(\partial_t v + \partial_y p) + \alpha \Delta y \, \partial_y \varphi(\partial_t p + \partial_x u + \partial_y v) = 0$$

$$\int \varphi(\partial_t p + \partial_x u + \partial_y v) + \alpha \Delta x \, \partial_x \varphi(\partial_t u + \partial_x p) + \alpha \Delta y \, \partial_y \varphi(\partial_t v + \partial_y p) = 0$$

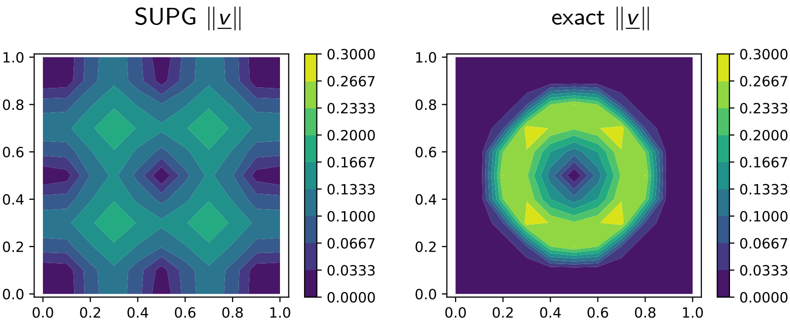
## Typical problems: SUPG

**Discretization**

- Cartesian grid
- CG-FEM
- SUPG
- $\mathbb{Q}^1$
- $N_x = 10$
- $N_y = 10$

**Test**

- Vortex $\underline{v}$
- $p \equiv 1$
- Long time simulation $T = 100$



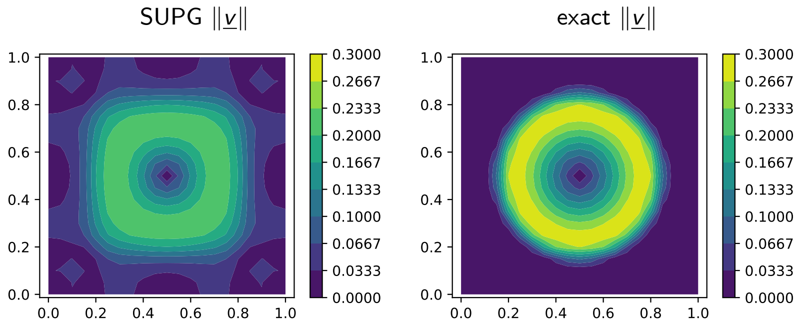
# Typical problems: SUPG

## Discretization

- Cartesian grid
- CG-FEM
- SUPG
- $\mathbb{Q}^1$
- $N_x = 20$
- $N_y = 20$

## Test

- Vortex $\underline{v}$
- $p \equiv 1$
- Long time simulation $T = 100$



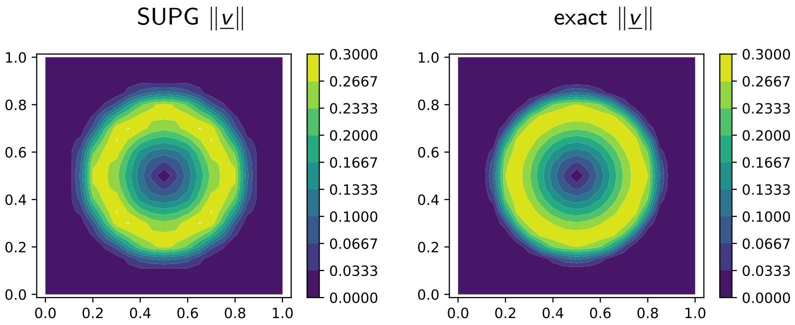
# Typical problems: SUPG

## Discretization

- Cartesian grid
- CG-FEM
- SUPG
- $\mathbb{Q}^2$
- $N_x = 10$
- $N_y = 10$

## Test

- Vortex $\underline{v}$
- $p \equiv 1$
- Long time simulation $T = 100$



# Typical problems: SUPG

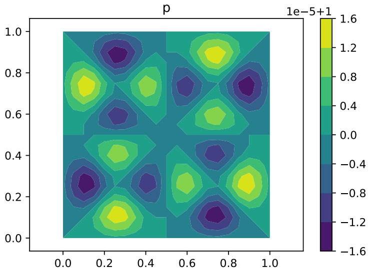
**Discretization**

- Cartesian grid
- CG-FEM
- SUPG

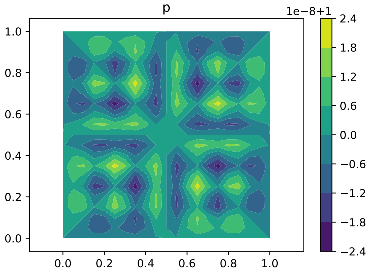
**Test**

- Vortex $\underline{v}$
- $p \equiv 1$
- Long time simulation $T = 100$

SUPG $p$, $\mathbb{Q}^1$, $N_x = N_y = 20$




exact $p$, $\mathbb{Q}^2$, $N_x = N_y = 10$



## Why also SUPG?

### Ideal

- At equilibrium $\partial_t q = 0$

### SUPG formulation

$$\int \varphi(\partial_t u + \partial_x p) + \alpha \Delta x \, \partial_x \varphi(\partial_t p + \partial_x u + \partial_y v) = 0$$

$$\int \varphi(\partial_t v + \partial_y p) + \alpha \Delta y \, \partial_y \varphi(\partial_t p + \partial_x u + \partial_y v) = 0$$

$$\int \varphi(\partial_t p + \partial_x u + \partial_y v) +$$

$$\alpha \Delta x \, \partial_x \varphi(\partial_t u + \partial_x p) + \alpha \Delta y \, \partial_y \varphi(\partial_t v + \partial_y p) = 0$$

# Why also SUPG?

## Ideal

- At equilibrium $\partial_t q = 0$
- $p \equiv c \in \mathbb{R}$
- $\nabla \cdot \underline{v} = 0$

## SUPG formulation

$$\int \varphi(\partial_t u + \partial_x p) + \alpha \Delta x \, \partial_x \varphi(\partial_t p + \partial_x u + \partial_y v) = 0$$

$$\int \varphi(\partial_t v + \partial_y p) + \alpha \Delta y \, \partial_y \varphi(\partial_t p + \partial_x u + \partial_y v) = 0$$

$$\int \varphi(\partial_t p + \partial_x u + \partial_y v) +$$
$$\alpha \Delta x \, \partial_x \varphi(\partial_t u + \partial_x p) + \alpha \Delta y \, \partial_y \varphi(\partial_t v + \partial_y p) = 0$$

# Why also SUPG?

## Ideal

- At equilibrium $\partial_t q = 0$
- $p \equiv c \in \mathbb{R}$
- $\nabla \cdot \underline{v} = 0$
- $\int \varphi \nabla p = 0 \quad \forall \varphi \in V_{h,0}^K$
- $\int \nabla \varphi \cdot \nabla p = 0 \quad \forall \varphi \in V_{h,0}^K$

## SUPG formulation

$$\int \varphi(\partial_t u + \partial_x p) + \alpha \Delta x \, \partial_x \varphi(\partial_t p + \partial_x u + \partial_y v) = 0$$

$$\int \varphi(\partial_t v + \partial_y p) + \alpha \Delta y \, \partial_y \varphi(\partial_t p + \partial_x u + \partial_y v) = 0$$

$$\int \varphi(\partial_t p + \partial_x u + \partial_y v) +$$

$$\alpha \Delta x \, \partial_x \varphi(\partial_t u + \partial_x p) + \alpha \Delta y \, \partial_y \varphi(\partial_t v + \partial_y p) = 0$$

# Why also SUPG?

## Ideal

- At equilibrium $\partial_t q = 0$
- $p \equiv c \in \mathbb{R}$
- $\nabla \cdot \underline{v} = 0$
- $\int \varphi \nabla p = 0 \quad \forall \varphi \in V_{h,0}^K$
- $\int \nabla \varphi \cdot \nabla p = 0 \quad \forall \varphi \in V_{h,0}^K$
- $\int \varphi \nabla \cdot \underline{v} = 0 \quad \forall \varphi \in V_{h,0}^K$
- $\int \nabla \varphi \nabla \cdot \underline{v} = 0 \quad \forall \varphi \in V_{h,0}^K$

## SUPG formulation

$$\int \varphi(\partial_t u + \partial_x p) + \alpha \Delta x \, \partial_x \varphi(\partial_t p + \partial_x u + \partial_y v) = 0$$

$$\int \varphi(\partial_t v + \partial_y p) + \alpha \Delta y \, \partial_y \varphi(\partial_t p + \partial_x u + \partial_y v) = 0$$

$$\int \varphi(\partial_t p + \partial_x u + \partial_y v) +$$

$$\alpha \Delta x \, \partial_x \varphi(\partial_t u + \partial_x p) + \alpha \Delta y \, \partial_y \varphi(\partial_t v + \partial_y p) = 0$$

# Why also SUPG?

## Ideal

- At equilibrium $\partial_t q = 0$
- $p \equiv c \in \mathbb{R}$
- $\nabla \cdot \underline{v} = 0$
- $\int \varphi \nabla p = 0 \quad \forall \varphi \in V_{h,0}^K$
- $\int \nabla\varphi \cdot \nabla p = 0 \quad \forall \varphi \in V_{h,0}^K$
- $\int \varphi \nabla \cdot \underline{v} = 0 \quad \forall \varphi \in V_{h,0}^K$
- $\int \nabla\varphi \nabla \cdot \underline{v} = 0 \quad \forall \varphi \in V_{h,0}^K$

## SUPG formulation

$$\int \varphi(\partial_t u + \partial_x p) + \alpha \Delta x\, \partial_x \varphi(\partial_t p + \partial_x u + \partial_y v) = 0$$

$$\int \varphi(\partial_t v + \partial_y p) + \alpha \Delta y\, \partial_y \varphi(\partial_t p + \partial_x u + \partial_y v) = 0$$

$$\int \varphi(\partial_t p + \partial_x u + \partial_y v) +$$

$$\alpha \Delta x\, \partial_x \varphi(\partial_t u + \partial_x p) + \alpha \Delta y\, \partial_y \varphi(\partial_t v + \partial_y p) = 0$$

## Practice

We just know that the combination of the operators is equal to 0.
Moreover, we would like to know the relation between the following

$$\ker\left[\int \varphi(\partial_x u + \partial_y v)\mathrm{d}x \quad \forall \varphi \in V_{h,0}^K\right] \quad \not\subset \quad \ker\left[\int \partial_x\varphi(\partial_x u + \partial_y v)\mathrm{d}x \quad \forall \varphi \in V_{h,0}^K\right].$$

## How to solve the problem?

### Recipe?

- 2D operators with more recognizable kernels
- Recast 2D operators to 1D operators to easily study their kernels
- Divergence operator that should be a Kronecher product of operators

## How to solve the problem?

### Recipe?

- 2D operators with more recognizable kernels
- Recast 2D operators to 1D operators to easily study their kernels
- Divergence operator that should be a Kronecher product of operators

### Global Flux

Reminder of what is Global Flux 1D (for balance laws)

$$\partial_t U + \partial_x F(U) = S(U)$$

$$G(U) := F(U) - \int^x S(U)$$

$$\partial_t U + \partial_x G(U) = 0$$

## How to solve the problem?

### Recipe?

- 2D operators with more recognizable kernels
- Recast 2D operators to 1D operators to easily study their kernels
- Divergence operator that should be a Kronecher product of operators

### Global Flux

Reminder of what is Global Flux 1D (for balance laws)

$$\partial_t U + \partial_x F(U) = S(U)$$

$$G(U) := F(U) - \int^x S(U)$$

$$\partial_t U + \partial_x G(U) = 0$$

### Global Flux SUPG for acoustics

Define $\boldsymbol{\sigma}_x(x,y) := \int_{y_0}^{y} u(x,s)ds$ and $\boldsymbol{\sigma}_y(x,y) := \int_{x_0}^{x} v(s,y)ds$, with $\boldsymbol{\sigma}_x, \boldsymbol{\sigma}_y \in V_h^K(\Omega_h)$, $\Phi := \boldsymbol{\sigma}_x + \boldsymbol{\sigma}_y$.

$$\int \varphi(\partial_t u + \partial_x p) + \alpha \Delta x \, \partial_x \varphi(\partial_t p + \partial_x \partial_y(\boldsymbol{\sigma}_x + \boldsymbol{\sigma}_y)) = 0$$

$$\int \varphi(\partial_t v + \partial_y p) + \alpha \Delta y \, \partial_y \varphi(\partial_t p + \partial_x \partial_y(\boldsymbol{\sigma}_x + \boldsymbol{\sigma}_y)) = 0$$

$$\int \varphi(\partial_t p + \partial_x \partial_y(\boldsymbol{\sigma}_x + \boldsymbol{\sigma}_y)) + \alpha \Delta x \, \partial_x \varphi(\partial_t u + \partial_x p) + \alpha \Delta y \, \partial_y \varphi(\partial_t v + \partial_y p) = 0$$

## How to solve the problem?

### Recipe?

- 2D operators with more recognizable kernels
- Recast 2D operators to 1D operators to easily study their kernels
- Divergence operator that should be a Kronecher product of operators

### Global Flux

Reminder of what is Global Flux 1D (for balance laws)

$$\partial_t U + \partial_x F(U) = S(U)$$

$$G(U) := F(U) - \int^x S(U)$$

$$\partial_t U + \partial_x G(U) = 0$$

### Global Flux SUPG for acoustics

Define $\boldsymbol{\sigma}_x(x,y) := \int_{y_0}^{y} u(x,s)ds$ and $\boldsymbol{\sigma}_y(x,y) := \int_{x_0}^{x} v(s,y)ds$, with $\boldsymbol{\sigma}_x, \boldsymbol{\sigma}_y \in V_h^K(\Omega_h)$, $\Phi := \boldsymbol{\sigma}_x + \boldsymbol{\sigma}_y$.

$$\int \varphi(\partial_t u + \partial_x p) + \alpha \Delta x\, \partial_x \varphi(\partial_t p + \partial_x \partial_y \Phi) = 0$$

$$\int \varphi(\partial_t v + \partial_y p) + \alpha \Delta y\, \partial_y \varphi(\partial_t p + \partial_x \partial_y \Phi) = 0$$

$$\int \varphi(\partial_t p + \partial_x \partial_y \Phi) + \alpha \Delta x\, \partial_x \varphi(\partial_t u + \partial_x p) + \alpha \Delta y\, \partial_y \varphi(\partial_t v + \partial_y p) = 0$$

# Global Flux SUPG

## Global Flux SUPG for acoustics

Define $\boldsymbol{\sigma}_x(x,y) := \int_{y_0}^{y} u(x,s)ds$ and $\boldsymbol{\sigma}_y(x,y) := \int_{x_0}^{x} v(s,y)ds$, with $\boldsymbol{\sigma}_x, \boldsymbol{\sigma}_y \in V_h^K(\Omega_h)$.

$$\int \varphi(\partial_t u + \partial_x p) + \alpha \Delta x\, \partial_x \varphi(\partial_t p + \partial_x \partial_y \Phi(u,v)) = 0$$

$$\int \varphi(\partial_t v + \partial_y p) + \alpha \Delta y\, \partial_y \varphi(\partial_t p + \partial_x \partial_y \Phi(u,v)) = 0$$

$$\int \varphi(\partial_t p + \partial_x \partial_y \Phi(u,v)) + \alpha \Delta x\, \partial_x \varphi(\partial_t u + \partial_x p) + \alpha \Delta y\, \partial_y \varphi(\partial_t v + \partial_y p) = 0$$

## Changes in equilibrium

$$\nabla \cdot \underline{v} = 0$$

$$\Longrightarrow \partial_x \partial_y (\boldsymbol{\sigma}_x + \boldsymbol{\sigma}_y) = 0$$

$$\Longleftrightarrow \boldsymbol{\sigma}_x + \boldsymbol{\sigma}_y = f(x) + g(y)$$

## Discrete operators on Cartesian grid

- $\int \partial_x \varphi \partial_x \partial_y \Phi = D_x^x \otimes D_y \Phi$
- $\int \partial_y \varphi \partial_x \partial_y \Phi = D_x \otimes D_y^y \Phi$
- $\int \varphi \partial_x \partial_y \Phi = D_x \otimes D_y \Phi$
- $\Phi_{i,j} := \int_{y_0}^{y_j} u\, dy + \int_{x_0}^{x_i} v\, dx$
- $(D_x)_{ij} = \int \phi_i(x) \partial_x \phi_j(x) dx$
- $(D_x^x)_{ij} = \int \partial_x \phi_i(x) \partial_x \phi_j(x) dx$
- $(I_x)_{ij} = \int_{x_0}^{x_i} \phi_j(x) dx$
- $\Phi = \mathrm{Id}_x \otimes I_y u + I_x \otimes \mathrm{Id}_y v$

# Properties summary

## Discretization

- $u, v, p, \sigma_x, \sigma_y, \Phi \in V_h^K$

# Properties summary

## Discretization

- $u, v, p, \sigma_x, \sigma_y, \Phi \in V_h^K$

## Equilibria

- $p \equiv C$ and $u, v$:

$$\Phi_{ij} = \int_{y_0}^{y_j} u(x_i, y)dy + \int_{x_0}^{x_i} v(x, y_j)dx$$

- $\Phi_{ij} = f_i + g_j$
- $\Phi_{ij} - \Phi_{i0} - \Phi_{0j} + \Phi_{00} = 0$ for all $i, j$

# Properties summary

## Discretization

- $u, v, p, \sigma_x, \sigma_y, \Phi \in V_h^K$

## Equilibria

- $p \equiv C$ and $u, v$:

$$\Phi_{ij} = \int_{y_0}^{y_j} u(x_i, y)dy + \int_{x_0}^{x_i} v(x, y_j)dx$$

- $\Phi_{ij} = f_i + g_j$
- $\Phi_{ij} - \Phi_{i0} - \Phi_{0j} + \Phi_{00} = 0$ for all $i, j$

## Dissipation of spurious modes

- Divergence operator $D_x \otimes D_y$ has spurious equilibria
- $D_x^x \otimes D_y$ or $D_x \otimes D_y^y$ dissipate essentially all spurious equilibria (we have a proof)

## Involution

- It is "possible" to compute the discrete involution, but not so nice

# FEM and quadrature

## FEM details

- Lagrangian basis functions
- Gauss–Lobatto nodes for quadrature
- Gauss–Lobatto nodes for basis function
- Tensor product/Kronecher product to 2D structures

## FEM and quadrature

### FEM details

- Lagrangian basis functions
- Gauss–Lobatto nodes for quadrature
- Gauss–Lobatto nodes for basis function
- Tensor product/Kronecher product to 2D structures

### Time Discretization

- Deferred Correction
- Arbitarily high order
- Explicit
- Diagonal mass matrix

## FEM and quadrature

### FEM details

- Lagrangian basis functions
- Gauss–Lobatto nodes for quadrature
- Gauss–Lobatto nodes for basis function
- Tensor product/Kronecher product to 2D structures

### Time Discretization

- Deferred Correction
- Arbitarily high order
- Explicit
- Diagonal mass matrix

### SUPG-GF FEM discretization

$$\Phi := \mathrm{Id}_x \otimes I_y\, u + I_x \otimes \mathrm{Id}_y\, v$$

$$0 = M_x \otimes M_y \partial_t u + D_x \otimes M_y p + \alpha \Delta x \left( D^x \otimes M_y \partial_t p + D_x^x \otimes D_y I_y u + D_x^x I_x \otimes D_y v \right),$$

$$0 = M_x \otimes M_y \partial_t v + M_x \otimes D_y p + \alpha \Delta y \left( M_x \otimes D^y \partial_t p + D_x \otimes D_y^y I_y u + D_x I_x \otimes D_y^y v \right),$$

$$0 = M_x \otimes M_y \partial_t p + D_x \otimes D_y I_y u + D_x I_x \otimes D_y v +$$

$$\alpha \left( \Delta x\, D^x \otimes M_y \partial_t u + \Delta y\, M_x \otimes D^y \partial_t v + (\Delta x\, D_x^x \otimes M_y + \Delta y\, M_x \otimes D_y^y) p \right).$$

## FEM and quadrature

### FEM details

- Lagrangian basis functions
- Gauss–Lobatto nodes for quadrature
- Gauss–Lobatto nodes for basis function
- Tensor product/Kronecher product to 2D structures

### Time Discretization

- Deferred Correction
- Arbitarily high order
- Explicit
- Diagonal mass matrix

### SUPG-GF FEM discretization

$$
\begin{aligned}
\Phi &:= \mathrm{Id}_x \otimes I_y\, u + I_x \otimes \mathrm{Id}_y\, v \\
0 =& M_x \otimes M_y \partial_t u + D_x \otimes M_y p + \alpha \Delta x \left( D^x \otimes M_y \partial_t p + D_x^x \otimes D_y \Phi(u,v) \right), \\
0 =& M_x \otimes M_y \partial_t v + M_x \otimes D_y p + \alpha \Delta y \left( M_x \otimes D^y \partial_t p + D_x \otimes D_y^y \Phi(u,v) \right), \\
0 =& M_x \otimes M_y \partial_t p + D_x \otimes D_y \Phi(u,v) + \\
& \alpha \left( \Delta x\, D^x \otimes M_y \partial_t u + \Delta y\, M_x \otimes D^y \partial_t v + (\Delta x\, D_x^x \otimes M_y + \Delta y\, M_x \otimes D_y^y) p \right).
\end{aligned}
$$

Safety check!

## Convergence of method on nonstationary problem with exact solution

$$\begin{cases} u(x,y,t) = -\frac{1}{2c}\left(\cos(\alpha\xi(x,y)+ct) - \cos(\alpha\xi(x,y)-ct)\right)\cos(\theta), \\ v(x,y,t) = -\frac{1}{2c}\left(\cos(\alpha\xi(x,y)+ct) - \cos(\alpha\xi(x,y)-ct)\right)\sin(\theta), \\ p(x,y,t) = \frac{1}{2}\left(\cos(\alpha\xi(x,y)+ct) + \cos(\alpha\xi(x,y)-ct)\right), \end{cases}$$

# Smooth nonstationary test: oblique flow

Safety check!

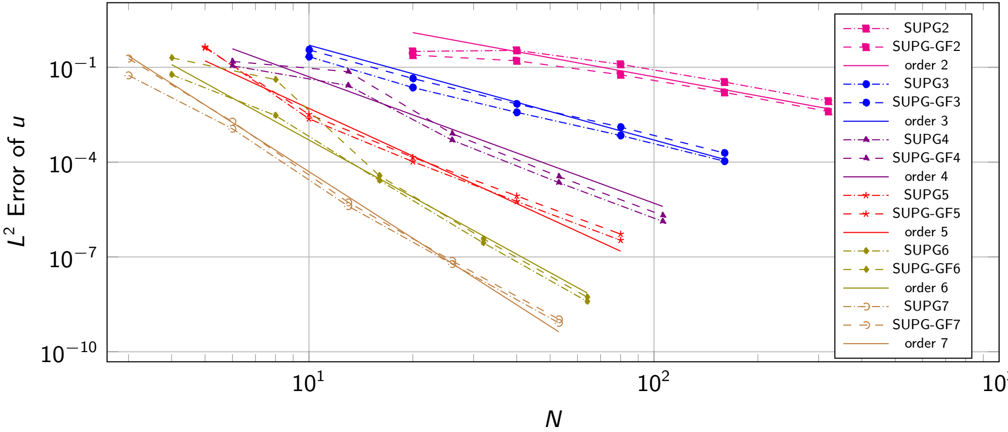



Figure: Oblique flow: convergence of $L^2$ error of $u$ with respect to the number of elements in $x$

## Simulation of vortex

$$\begin{cases} u(x,y) = f(\rho(x,y) \cdot (y - y_0) \\ v(x,y) = -f(\rho(x,y)) \cdot (x - x_0) \\ p(x,y) = 1 \end{cases}$$

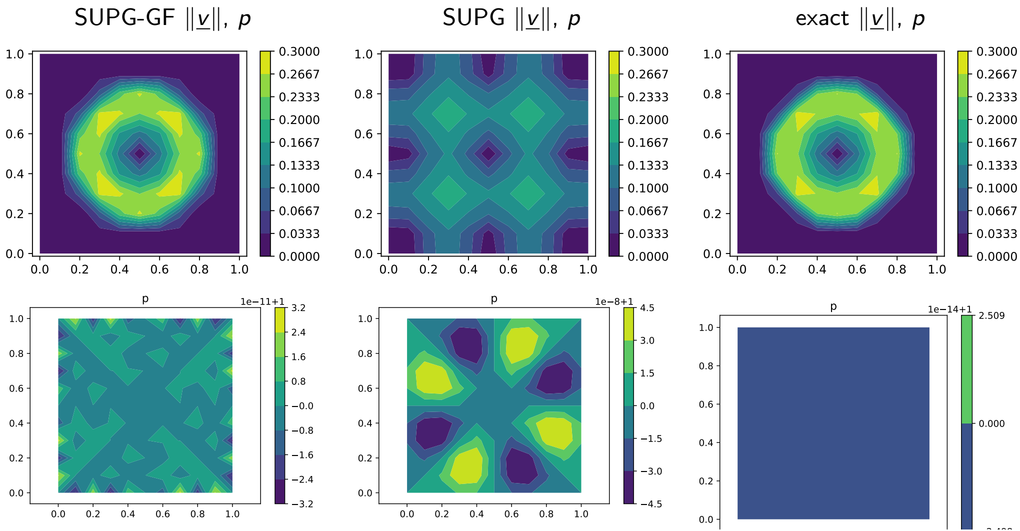
with $\rho(x,y) = \frac{\sqrt{(x-x_0)^2+(y-y_0)^2}}{r_0}$ with $r_0 = 0.45$ the radius of the support.

$$f(\rho) = 2\gamma e^{-\frac{1}{2(1-\rho)^2}} \sqrt{\frac{g}{r_0(1-\rho)^3}}$$

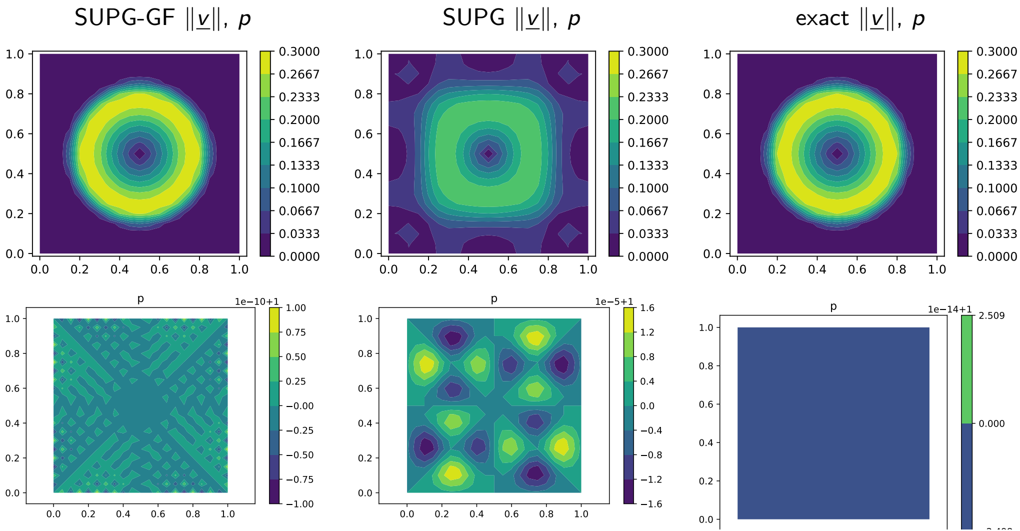
with $g = 9.81$, $\gamma = 0.2$ if $\rho < 1$, else 0.

$$T = 100$$

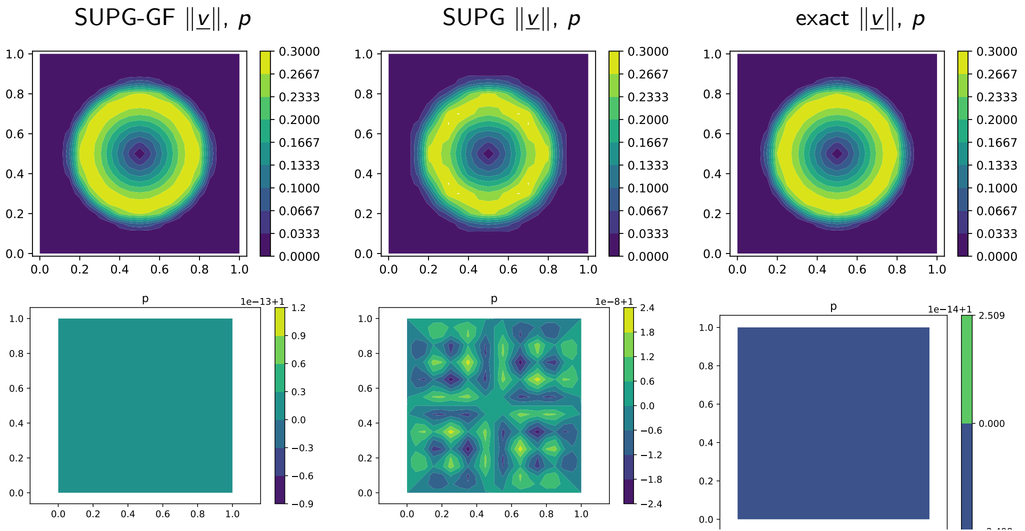
# Simulation of vortex: $\mathbb{Q}^1$, $N_x = N_y = 10$



# Simulation of vortex: $\mathbb{Q}^1$, $N_x = N_y = 20$



# Simulation of vortex: $\mathbb{Q}^2, \quad N_x = N_y = 10$



# Simulation of vortex: errors

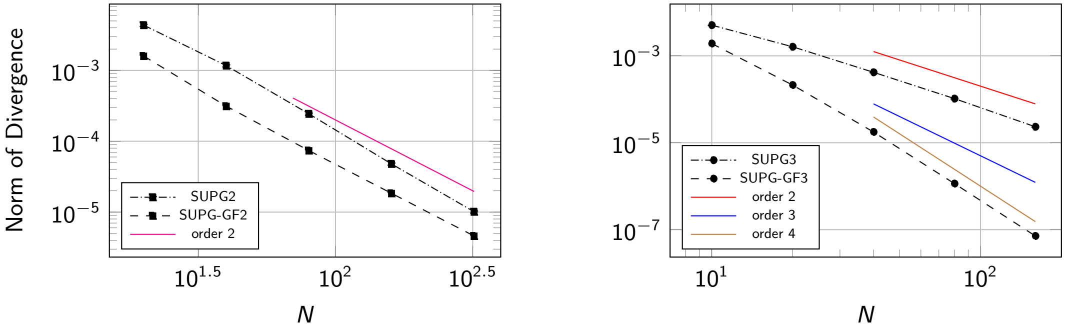



Figure: Smooth vortex: convergence of $L^2$ error of $u$ with respect to the number of elements in $x$

# Simulation of vortex: errors

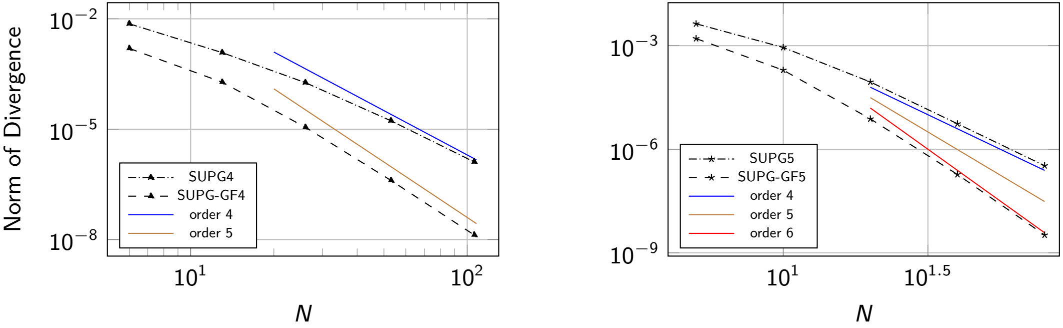



Figure: Smooth vortex: convergence of $L^2$ error of $u$ with respect to the number of elements in $x$

## Vortex simulation: divergence error

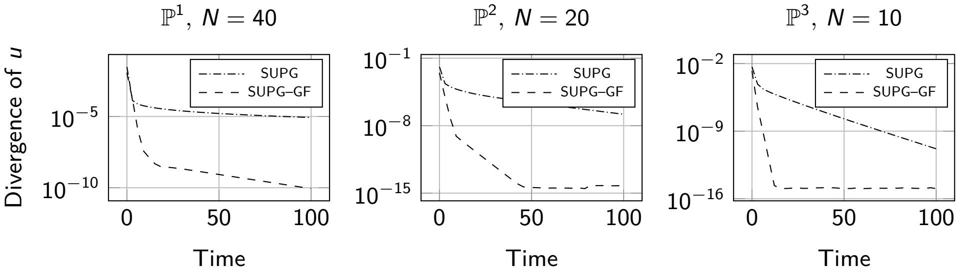



Figure: Norm of discrete divergence of $\underline{u}$ for SUPG $(\partial_x u + \partial_y v)$ and SUPG–GF $(\partial_x \partial_y (\boldsymbol{\sigma}_x + \boldsymbol{\sigma}_y))$ simulations with respect to time for different orders

# Vortex perturbation

## Pressure perturbation

- Gaussian centered in $\underline{x}_p = (0.4, 0.43)$
- scaling coefficient $r_0 = 0.1$
- radius $\rho(\underline{x}) = \sqrt{||\underline{x} - \underline{x}_p||}/r_0$

$$\delta_p(\underline{x}) = \varepsilon e^{-\frac{1}{2(1-\rho(\underline{x}))^2}+\frac{1}{2}},$$

- final time $T = 0.35$

## Vortex perturbation

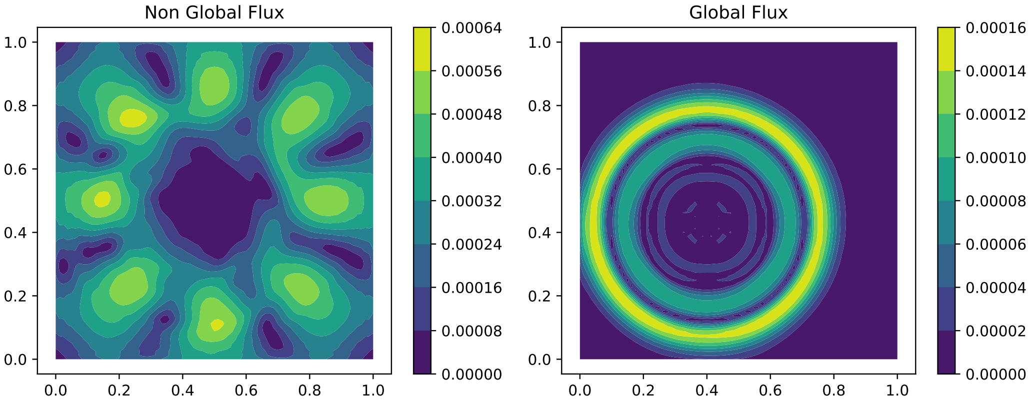



Figure: Perturbation($\varepsilon = 10^{-3}$) test. Plot of $\|\underline{u}_{eq} - \underline{u}_p\|$, with $\underline{u}_{eq}$ the equilibrium obtained with a cheap optimization process. $\mathbb{P}^1$ with $80 \times 80$ cells and 6561 dofs.

# Vortex perturbation

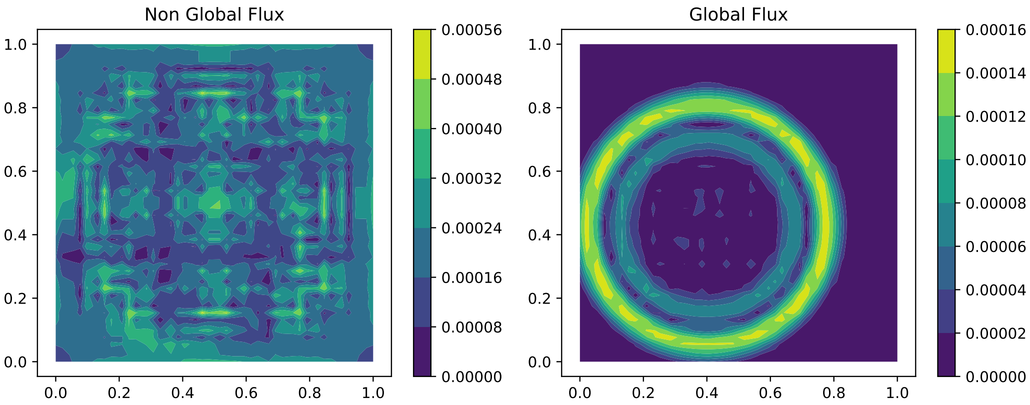



Figure: Perturbation($\varepsilon = 10^{-3}$) test. Plot of $\|\underline{u}_{eq} - \underline{u}_p\|$, with $\underline{u}_{eq}$ the equilibrium obtained with a cheap optimization process. $\mathbb{P}^3$ with $13 \times 13$ cells and 1600 dofs.

## Vortex perturbation

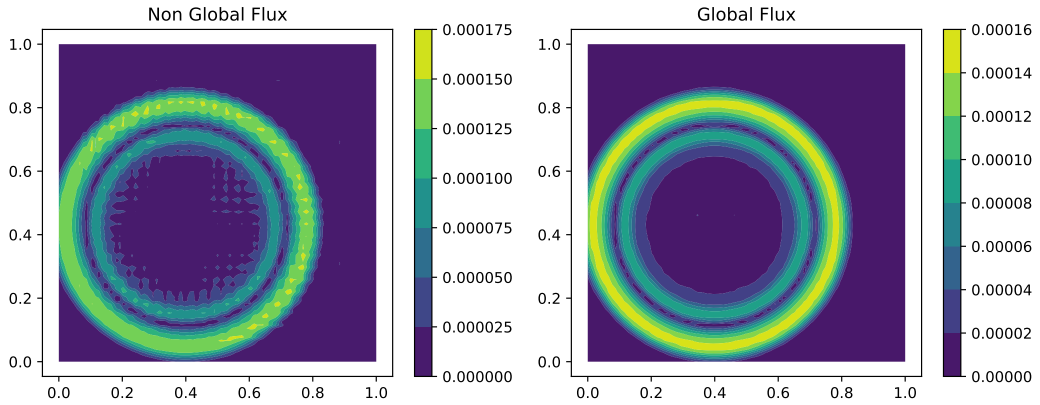



Figure: Perturbation($\varepsilon = 10^{-3}$) test. Plot of $\|\underline{u}_{eq} - \underline{u}_p\|$, with $\underline{u}_{eq}$ the equilibrium obtained with a cheap optimization process. $\mathbb{P}^3$ with $26 \times 26$ cells and 6241 dofs.

## Other models

### Acoustic with Coriolis

$$\partial_t \begin{pmatrix} u \\ v \\ p \end{pmatrix} + \partial_x \begin{pmatrix} p \\ 0 \\ u \end{pmatrix} + \partial_y \begin{pmatrix} 0 \\ p \\ v \end{pmatrix} + c_f \begin{pmatrix} -v \\ u \\ 0 \end{pmatrix} = 0.$$

### Acoustic with source term

$$\partial_t \begin{pmatrix} u \\ v \\ p \end{pmatrix} + \partial_x \begin{pmatrix} p \\ 0 \\ u \end{pmatrix} + \partial_y \begin{pmatrix} 0 \\ p \\ v \end{pmatrix} + \begin{pmatrix} 0 \\ 0 \\ s \end{pmatrix} = 0.$$

### Stommel Gyre

$$\partial_t p = -\operatorname{div}\mathbf{u}$$
$$\partial_t \mathbf{u} = -\operatorname{grad} p + \phi \mathbf{u}^\perp - R\mathbf{u} + \boldsymbol{\tau}$$

## Extensions

### Model

- Other stabilizations (OSS, CIP)
- Other equations

### Triangular meshes

- Haven't tried yet
- In principle, we can still define $\Phi : \int^y u + \int^x v$ in each element
- Question: will it be that effective?
- Kernels? Maybe difficult to write, still working

# THANKS!!

## References

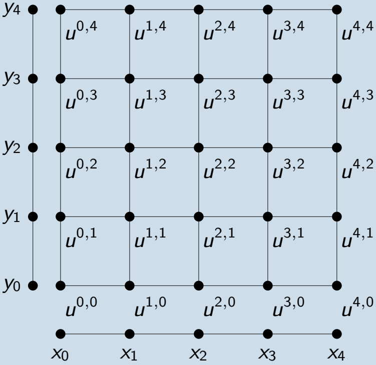

- Y. Cheng, A. Chertock, M. Herty, A. Kurganov and T. Wu. *A new approach for designing moving-water equilibria preserving schemes for the shallow water equations.* J. Sci. Comput. 80(1): 538–554, 2019.
- M. Ciallella, D. Torlo and M. Ricchiuto. *Arbitrary high order WENO finite volume scheme with flux globalization for moving equilibria preservation.* Journal of Scientific Computing, 96(2):53, 2023.
- W. Barsukow, M. Ricchiuto and D. Torlo. *Structure preserving methods via Global Flux quadrature: divergence-free preservation with continuous Finite Element.* In preparation, 2024.

- davidetorlo.it

# Detailed definition of Global Flux SUPG

## Definition of $\boldsymbol{\sigma}_x$, $\boldsymbol{\sigma}_y$

Cartesian grid, Lagrangian basis functions in Lobatto points $(x_i, y_j)$ in each direction.
So, $\phi_i(x_k) = \delta_{ik}$ and $\psi_j(y_\ell) = \delta_{j\ell}$ and




$$u(x,y) = \sum_{i,j} \varphi_{ij}(x,y) u^{i,j} = \sum_{i,j} \phi_i(x)\psi_j(y) u^{ij}$$

$$u(x_i, y_j) = u^{i,j}$$

$$\boldsymbol{\sigma}_x(x,y) = \sum_{i,j} \phi_i(x)\psi_j(y)\boldsymbol{\sigma}_x^{i,j}$$

$$\boldsymbol{\sigma}_x(x_i, y_j) = \boldsymbol{\sigma}_x^{i,j}$$

$$\boldsymbol{\sigma}_x(x,y) = \int_{y_0}^{y} u(x,s)ds$$

$$\boldsymbol{\sigma}_x^{i,j} = U(x_i, y_j) = \int_{y_0}^{y_j} u(x_i, s)ds = \sum_{k,\ell} \phi_k(x_i) \int_{y_0}^{y_j} \psi_\ell(s)ds\, u^{k,\ell}$$

So, even if both $\boldsymbol{\sigma}_x, u \in V_h^K$, in quadrature points, we have that exactly $u(x_i, y_j) = \int_{y_0}^{y_j} \boldsymbol{\sigma}_x(x_i, y)dy$.

### Global Flux is not global!

- In principle $\boldsymbol{\sigma}_x(x,y) = \int_{y_B}^{y} u(x,s)ds$ should be integrated from the beginning (bottom) of the domain $y_B$!
- In practice we always use $\partial_x \partial_y \boldsymbol{\sigma}_x(x,y)$ integrated in one cell!!!!
- So,

$$\boldsymbol{\sigma}_x(x,y) = \int_{y_B}^{y} u(x,s)ds = \underbrace{\int_{y_B}^{y_0} u(x,s)ds}_{\text{constant in one cell!}} + \int_{y_0}^{y} u(x,s)ds$$

  whatever constant we bring from outside the cell, is canceled out

$$\partial_y \boldsymbol{\sigma}_x(x,y) = \partial_y \int_{y_B}^{y} u(x,s)ds = \partial_y \int_{y_B}^{y_0} u(x,s)ds + \partial_y \int_{y_0}^{y} u(x,s)ds = \partial_y \int_{y_0}^{y} u(x,s)ds$$

# Why SUPG-GF works so better?

## Clearly divergence-free preserving

- Which divergence? $\partial_x \partial_y (\boldsymbol{\sigma}_x + \boldsymbol{\sigma}_y) \approx \partial_x \partial_y \left( \int^y u(x,s) ds + \int^x v(s,y) ds \right) = \partial_x u + \partial_y v$

## Why SUPG-GF works so better?

### Clearly divergence-free preserving

- Which divergence? $\partial_x \partial_y (\boldsymbol{\sigma}_x + \boldsymbol{\sigma}_y) \approx \partial_x \partial_y \left( \int^y u(x,s)ds + \int^x v(s,y)ds \right) = \partial_x u + \partial_y v$

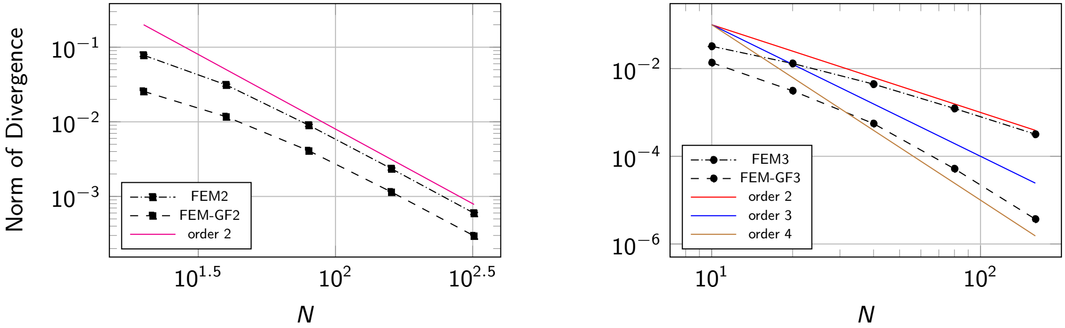



Figure: Smooth vortex: convergence of divergence operator on exact IC with respect to the number of elements in $x$

# Why SUPG-GF works so better?

## Clearly divergence-free preserving

- Which divergence? $\partial_x \partial_y (\boldsymbol{\sigma}_x + \boldsymbol{\sigma}_y) \approx \partial_x \partial_y \left( \int^y u(x,s)ds + \int^x v(s,y)ds \right) = \partial_x u + \partial_y v$

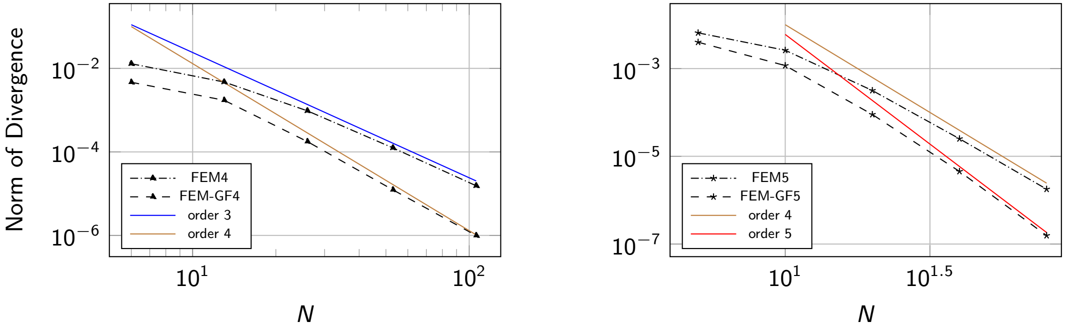



Figure: Smooth vortex: convergence of divergence operator on exact IC with respect to the number of elements in $x$

## Why SUPG-GF works so better?

### Clearly divergence-free preserving

- Which divergence? $\partial_x \partial_y (\boldsymbol{\sigma}_x + \boldsymbol{\sigma}_y) \approx \partial_x \partial_y \left( \int^y u(x,s)ds + \int^x v(s,y)ds \right) = \partial_x u + \partial_y v$

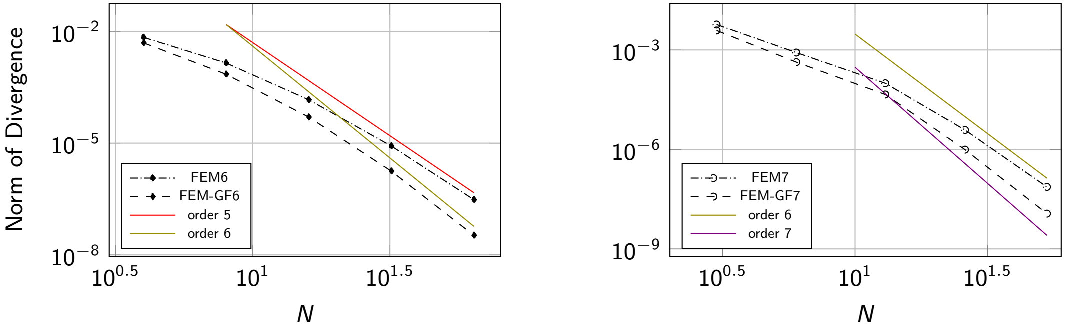



Figure: Smooth vortex: convergence of divergence operator on exact IC with respect to the number of elements in $x$

# Why SUPG-GF works so better?

## Clearly divergence-free preserving

- Which divergence? $\partial_x\partial_y(\boldsymbol{\sigma}_x+\boldsymbol{\sigma}_y) \approx \partial_x\partial_y\left(\int^y u(x,s)ds + \int^x v(s,y)ds\right) = \partial_x u + \partial_y v$
- If we know that $\partial_x\partial_y(\boldsymbol{\sigma}_x+\boldsymbol{\sigma}_y)=0$ and $p\equiv c$ then equilibrium

## Why SUPG-GF works so better?

### New operators kernels

$$\Phi = \boldsymbol{\sigma}_x + \boldsymbol{\sigma}_y$$

$$\int_{\Omega_h} \varphi(x,y)\partial_x\partial_y(\Phi)dxdy = 0 \qquad \forall \varphi \in V^K_{h,0}$$

$$\int_{\Omega_h} \partial_x\varphi(x,y)\partial_x\partial_y(\Phi)dxdy = 0 \qquad \forall \varphi \in V^K_{h,0}$$

$$\int_{\Omega_h} \partial_y\varphi(x,y)\partial_x\partial_y(\Phi)dxdy = 0 \qquad \forall \varphi \in V^K_{h,0}$$

### Matrix formulation

$$(D_x)_{ij} := \int \phi_i(x)\partial_x\phi_j(x)dx \qquad (D^x_x)_{ij} := \int \partial_x\phi_i(x)\partial_x\phi_j(x)dx$$

$$\Phi = \boldsymbol{\sigma}_x + \boldsymbol{\sigma}_y \qquad \Phi \in \mathbb{R}^{(N_xK+1)\times(N_yK+1)}$$

$$(D_x \otimes D_y)\,\Phi = 0 \qquad D_x, D^x_x \in \mathbb{R}^{(N_xK-1)\times(N_xK+1)}$$

$$(D^x_x \otimes D_y)\,\Phi = 0 \qquad D_y, D^y_y \in \mathbb{R}^{(N_yK-1)\times(N_yK+1)}$$

$$(D_x \otimes D^y_y)\,\Phi = 0$$

### Kernels of Kronecher products

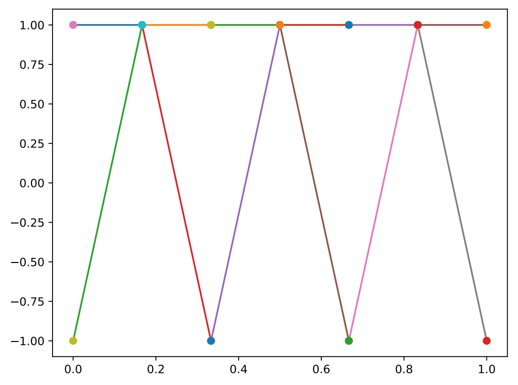
$M_x \otimes M_y \Phi = 0 \iff$

$M_x \Phi^{:,j} = 0\, \forall j$ or $M_y \Phi^{i,:} = 0\, \forall i$

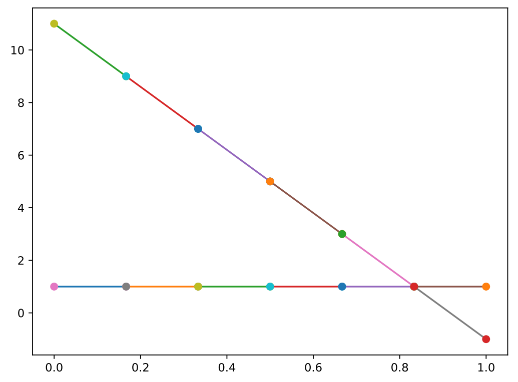
We can pass from the study of the 2D operators to the 1D operators!
Reminder: before it was not possible because we had a combination of operators $D_x u + D_y v = 0$.

# One dimensional kernels of $D_x$ and $D_x^x$

## Kernel of $D_x$




## Kernel of $D_x^x$

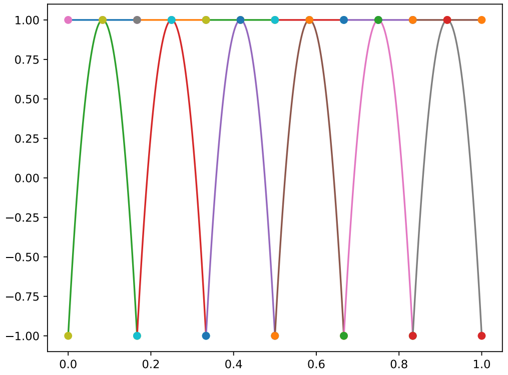



## Operators

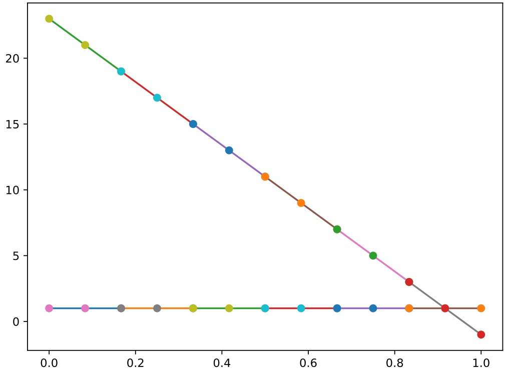
- Divergence $D_x \otimes D_y$
- Stabilization $D_x^x \otimes D_y$, $D_x \otimes D_y^y$

# One dimensional kernels of $D_x$ and $D_x^x$

## Kernel of $D_x$




## Kernel of $D_x^x$

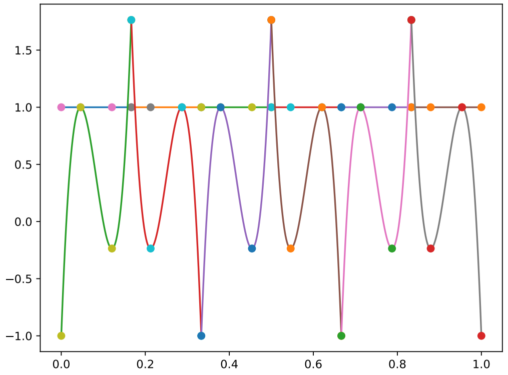



## Operators

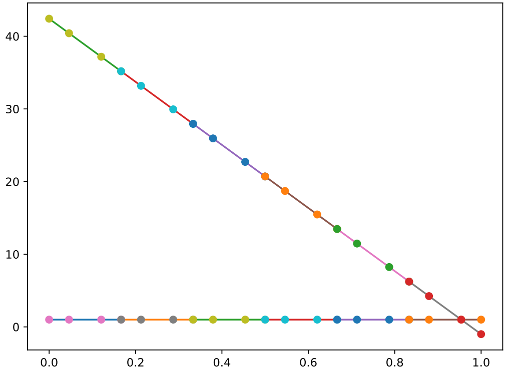
- Divergence $D_x \otimes D_y$
- Stabilization $D_x^x \otimes D_y$, $D_x \otimes D_y^y$

## One dimensional kernels of $D_x$ and $D_x^x$

### Kernel of $D_x$




### Kernel of $D_x^x$




### Operators

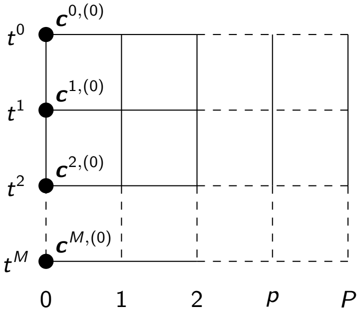
- Divergence $D_x \otimes D_y$
- Stabilization $D_x^x \otimes D_y$, $D_x \otimes D_y^y$

## Deferred Correction Iterative procedure

How to combine two methods keeping the accuracy of the second and the stability and simplicity of the first one?

$$\mathbf{c}^{0,(p)} := \mathbf{c}(t_n), \quad p = 0, \dots, P,$$
$$\mathbf{c}^{m,(0)} := \mathbf{c}(t_n), \quad m = 1, \dots, M$$
$$T^1(\underline{\mathbf{c}}^{(p)}) = T^1(\underline{\mathbf{c}}^{(p-1)}) - T^2(\underline{\mathbf{c}}^{(p-1)}) \text{ with } p = 1, \dots, P.$$

- $T^1(\underline{\mathbf{c}}) = 0$, first order accuracy, easily invertible.
- $T^2(\underline{\mathbf{c}}) = 0$, high order $Q$.




### DeC Theorem

- $T^1$ coercive with constant $\mathcal{O}(1)$
- $T^1 - T^2$ Lipschitz with constant $\mathcal{O}(\Delta t)$

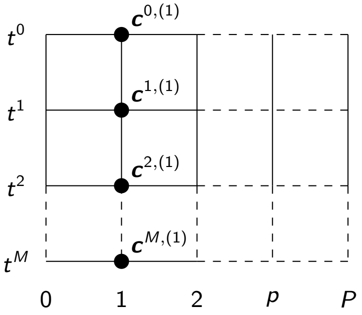
DeC converges and $\min(P, Q)$ is the order of accuracy.

## Deferred Correction Iterative procedure

How to combine two methods keeping the accuracy of the second and the stability and simplicity of the first one?

$$\mathbf{c}^{0,(p)} := \mathbf{c}(t_n), \quad p = 0, \dots, P,$$
$$\mathbf{c}^{m,(0)} := \mathbf{c}(t_n), \quad m = 1, \dots, M$$
$$T^1(\underline{\mathbf{c}}^{(p)}) = T^1(\underline{\mathbf{c}}^{(p-1)}) - T^2(\underline{\mathbf{c}}^{(p-1)}) \text{ with } p = 1, \dots, P.$$

- $T^1(\underline{\mathbf{c}}) = 0$, first order accuracy, easily invertible.
- $T^2(\underline{\mathbf{c}}) = 0$, high order $Q$.




### DeC Theorem

- $T^1$ coercive with constant $\mathcal{O}(1)$
- $T^1 - T^2$ Lipschitz with constant $\mathcal{O}(\Delta t)$

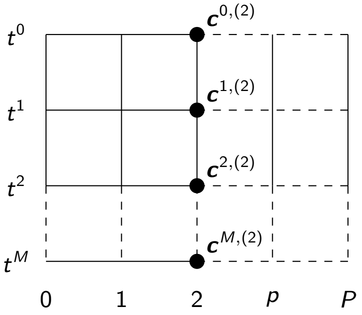
DeC converges and $\min(P, Q)$ is the order of accuracy.

## Deferred Correction Iterative procedure

How to combine two methods keeping the accuracy of the second and the stability and simplicity of the first one?

- $T^1(\underline{\mathbf{c}}) = 0$, first order accuracy, easily invertible.
- $T^2(\underline{\mathbf{c}}) = 0$, high order $Q$.

$$\mathbf{c}^{0,(p)} := \mathbf{c}(t_n), \quad p = 0, \dots, P,$$
$$\mathbf{c}^{m,(0)} := \mathbf{c}(t_n), \quad m = 1, \dots, M$$
$$T^1(\underline{\mathbf{c}}^{(p)}) = T^1(\underline{\mathbf{c}}^{(p-1)}) - T^2(\underline{\mathbf{c}}^{(p-1)}) \text{ with } p = 1, \dots, P.$$




### DeC Theorem

- $T^1$ coercive with constant $\mathcal{O}(1)$
- $T^1 - T^2$ Lipschitz with constant $\mathcal{O}(\Delta t)$

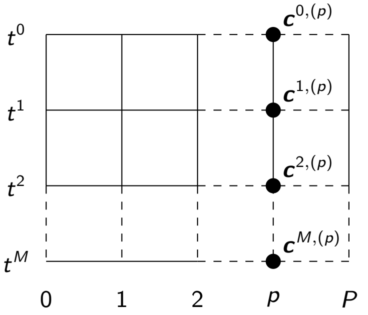
DeC converges and $\min(P, Q)$ is the order of accuracy.

## Deferred Correction Iterative procedure

How to combine two methods keeping the accuracy of the second and the stability and simplicity of the first one?

- $T^1(\underline{\boldsymbol{c}}) = 0$, first order accuracy, easily invertible.
- $T^2(\underline{\boldsymbol{c}}) = 0$, high order $Q$.

$$\boldsymbol{c}^{0,(p)} := \boldsymbol{c}(t_n), \quad p = 0, \dots, P,$$
$$\boldsymbol{c}^{m,(0)} := \boldsymbol{c}(t_n), \quad m = 1, \dots, M$$
$$T^1(\underline{\boldsymbol{c}}^{(p)}) = T^1(\underline{\boldsymbol{c}}^{(p-1)}) - T^2(\underline{\boldsymbol{c}}^{(p-1)}) \text{ with } p = 1, \dots, P.$$




### DeC Theorem

- $T^1$ coercive with constant $\mathcal{O}(1)$
- $T^1 - T^2$ Lipschitz with constant $\mathcal{O}(\Delta t)$

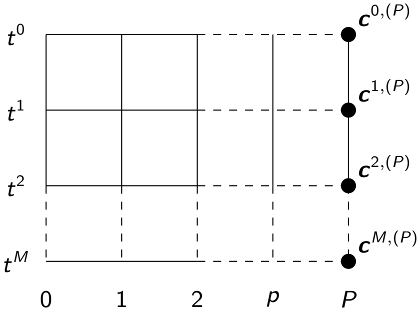
DeC converges and $\min(P, Q)$ is the order of accuracy.

## Deferred Correction Iterative procedure

How to combine two methods keeping the accuracy of the second and the stability and simplicity of the first one?

- $T^1(\underline{\mathbf{c}}) = 0$, first order accuracy, easily invertible.
- $T^2(\underline{\mathbf{c}}) = 0$, high order $Q$.

$$\mathbf{c}^{0,(p)} := \mathbf{c}(t_n), \quad p = 0, \dots, P,$$
$$\mathbf{c}^{m,(0)} := \mathbf{c}(t_n), \quad m = 1, \dots, M$$
$$T^1(\underline{\mathbf{c}}^{(p)}) = T^1(\underline{\mathbf{c}}^{(p-1)}) - T^2(\underline{\mathbf{c}}^{(p-1)}) \text{ with } p = 1, \dots, P.$$




### DeC Theorem

- $T^1$ coercive with constant $\mathcal{O}(1)$
- $T^1 - T^2$ Lipschitz with constant $\mathcal{O}(\Delta t)$

DeC converges and $\min(P, Q)$ is the order of accuracy.

# DeC for SUPG

$$
\begin{aligned}
T_u^{2,m}(\underline{\underline{q}}) =& M_x \otimes M_y \frac{u^m - u^0}{\Delta t} + D_x \otimes M_y \sum_r \theta_r^m p^r + \\
& \alpha h (D^x \otimes M_y \frac{p^m - p^0}{\Delta t} + D_x^x \otimes D_y I_y \sum_r \theta_r^m u^r + D_x^x I_x \otimes D_y \sum_r \theta_r^m v^r ),
\end{aligned} \tag{3a}
$$

$$
\begin{aligned}
T_v^{2,m}(\underline{\underline{q}}) =& M_x \otimes M_y \frac{v^m - v^0}{\Delta t} + M_x \otimes D_y \sum_r \theta_r^m p^r + \\
& \alpha h (M_x \otimes D^y \frac{p^m - p^0}{\Delta t} + D_x \otimes D_y^y I_y \sum_r \theta_r^m u^r + D_x I_x \otimes D_y^y \sum_r \theta_r^m v^r ),
\end{aligned} \tag{3b}
$$

$$
\begin{aligned}
T_p^{2,m}(\underline{\underline{q}}) =& M_x \otimes M_y \frac{p^m - p^0}{\Delta t} + D_x \otimes D_y I_y \sum_r \theta_r^m u^r + D_x I_x \otimes D_y \sum_r \theta_r^m v^r + \\
& \alpha h (D^x \otimes M_y \frac{u^m - u^0}{\Delta t} + M_x \otimes D^y \frac{v^m - v^0}{\Delta t} + (D_x^x \otimes M_y + M_x \otimes D_y^y) \sum_r \theta_r^m p^r ).
\end{aligned} \tag{3c}
$$

# DeC for SUPG

$$T_u^{1,m}(\underline{\underline{q}}) = M_x \otimes M_y \frac{u^m - u^0}{\Delta t} + \beta^m D_x \otimes M_y p^0, \tag{2a}$$

$$T_v^{1,m}(\underline{\underline{q}}) = M_x \otimes M_y \frac{v^m - v^0}{\Delta t} + \beta^m M_x \otimes D_y p^0, \tag{2b}$$

$$T_p^{1,m}(\underline{\underline{q}}) = M_x \otimes M_y \frac{p^m - p^0}{\Delta t} + \beta^m (D_x \otimes D_y I_y u^0 + D_x I_x \otimes D_y v^0). \tag{2c}$$

$$\begin{aligned} T_u^{2,m}(\underline{\underline{q}}) = & M_x \otimes M_y \frac{u^m - u^0}{\Delta t} + D_x \otimes M_y \sum_r \theta_r^m p^r + \\ & \alpha h (D^x \otimes M_y \frac{p^m - p^0}{\Delta t} + D_x^x \otimes D_y I_y \sum_r \theta_r^m u^r + D_x^x I_x \otimes D_y \sum_r \theta_r^m v^r), \end{aligned} \tag{3a}$$

$$\begin{aligned} T_v^{2,m}(\underline{\underline{q}}) = & M_x \otimes M_y \frac{v^m - v^0}{\Delta t} + M_x \otimes D_y \sum_r \theta_r^m p^r + \\ & \alpha h (M_x \otimes D^y \frac{p^m - p^0}{\Delta t} + D_x \otimes D_y^y I_y \sum_r \theta_r^m u^r + D_x I_x \otimes D_y^y \sum_r \theta_r^m v^r), \end{aligned} \tag{3b}$$

$$\begin{aligned} T_p^{2,m}(\underline{\underline{q}}) = & M_x \otimes M_y \frac{p^m - p^0}{\Delta t} + D_x \otimes D_y I_y \sum_r \theta_r^m u^r + D_x I_x \otimes D_y \sum_r \theta_r^m v^r + \\ & \alpha h (D^x \otimes M_y \frac{u^m - u^0}{\Delta t} + M_x \otimes D^y \frac{v^m - v^0}{\Delta t} + (D_x^x \otimes M_y + M_x \otimes D_y^y) \sum_r \theta_r^m p^r). \end{aligned} \tag{3c}$$

## Vortex with Coriolis

### Acoustic with Coriolis

$$\partial_t \begin{pmatrix} u \\ v \\ p \end{pmatrix} + \partial_x \begin{pmatrix} p \\ 0 \\ u \end{pmatrix} + \partial_y \begin{pmatrix} 0 \\ p \\ v \end{pmatrix} + c_f \begin{pmatrix} -v \\ u \\ 0 \end{pmatrix} = 0.$$

### GF for Acoustic with Coriolis

$$\partial_t \begin{pmatrix} u \\ v \\ p \end{pmatrix} + \begin{pmatrix} \partial_x(p + \boldsymbol{\sigma}_y) \\ \partial_y(p - \boldsymbol{\sigma}_x) \\ \partial_x\partial_y(\boldsymbol{\sigma}_x + \boldsymbol{\sigma}_y) \end{pmatrix} = 0$$

### FEM change

$$T_u^{2,m}(\underline{\underline{q}}) {+}{=} {-}c_f M_x \otimes M_y v^m$$

$$T_v^{2,m}(\underline{\underline{q}}) {+}{=} c_f M_x \otimes M_y u^m$$

$$T_p^{2,m}(\underline{\underline{q}}) {+}{=} c_f \alpha \sum_r \theta_r^m \left( M_x \otimes D^y u^r - D^x \otimes M_y v^r \right)$$

### GF-FEM change

$$T_u^{2,m}(\underline{\underline{q}}) {+}{=} {-}c_f D_x I_x \otimes M_x v^m$$

$$T_v^{2,m}(\underline{\underline{q}}) {+}{=} c_f M_x \otimes D_y I_y u^m$$

$$T_p^{2,m}(\underline{\underline{q}}) {+}{=} c_f \alpha \sum_r \theta_r^m \left( M_x \otimes D_y^y I_y u^r - D_x^x I_x \otimes M_y v^r \right)$$

# Vortex with Coriolis

## Test

- $$\begin{cases} u(x,y) = -f(\rho(x,y)) \cdot (y - y_0), \\ v(x,y) = f(\rho(x,y)) \cdot (x - x_0), \\ p(x,y) = 1 - c_f \cdot g(\rho(x,y)), \end{cases}$$
- $\rho(x,y) = \sqrt{x^2 + y^2}$
- $f(\rho) := 20 e^{-100\rho^2}$
- $g(\rho) := \frac{1}{10} e^{-100\rho^2}$
- Domain $\Omega = [0,1]^2$

## Vortex with Coriolis

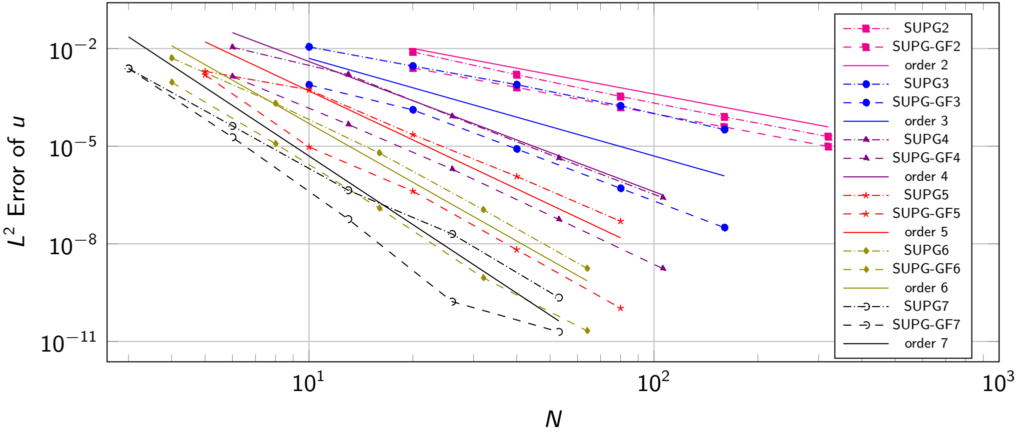
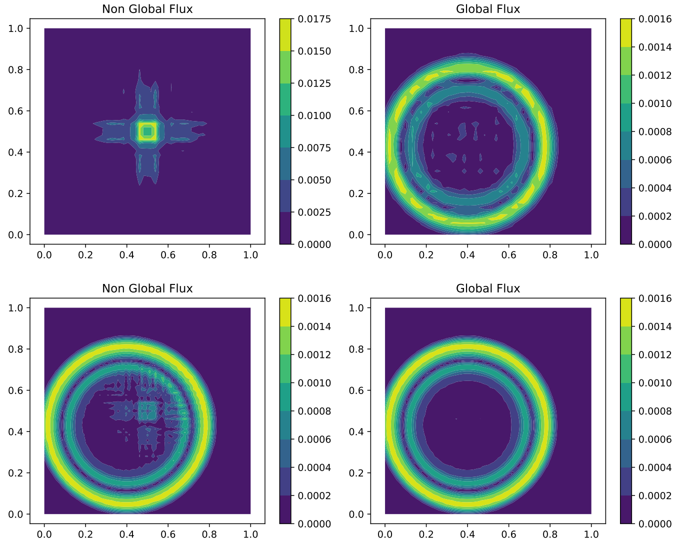


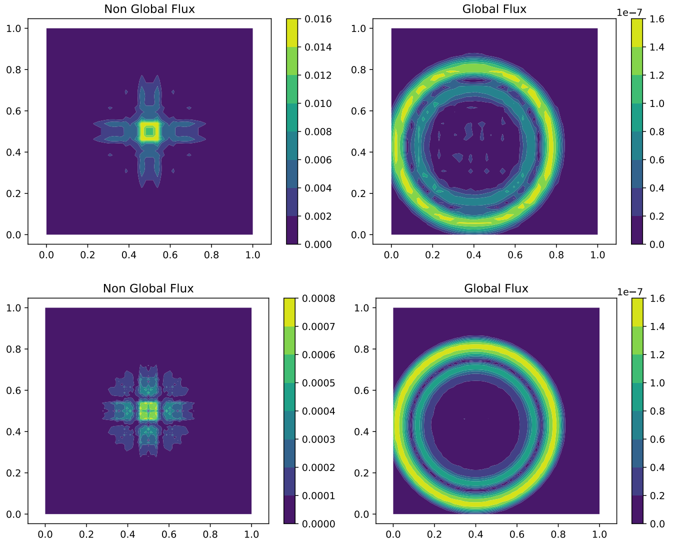

Figure: Coriolis vortex: convergence of $L^2$ error of $u$ with respect to the number of elements in $x$

# Vortex with Coriolis




Perturbation($\varepsilon = 10^{-2}$) test.
Plot of $\|\underline{u}_{eq} - \underline{u}_p\|$, with $\underline{u}_{eq}$ the analytical equilibrium.
Top $\mathbb{P}^3$ with 13 cells, bottom $\mathbb{P}^3$ with 26 cells.

# Vortex with Coriolis




Perturbation($\varepsilon = 10^{-6}$) test.
Plot of $\|\underline{u}_{eq} - \underline{u}_p\|$, with $\underline{u}_{eq}$ the analytical equilibrium.
Top $\mathbb{P}^3$ with 13 cells, bottom $\mathbb{P}^3$ with 26 cells.

## Source term

Consider the source equations

$$\begin{cases} \partial_t \underline{u} + \nabla p = 0, \\ \partial_t p + \nabla \cdot \underline{u} = s, \end{cases} \tag{4}$$

where an equilibrium solution can be found as

$$\begin{cases} p(x,y) \equiv p_0 \in \mathbb{R}, \\ \underline{u}(x,y) = \nabla^{\perp}\phi_1(x,y) + \nabla \phi_2(x,y), \\ s(x,y) = \Delta \phi_2(x,y), \end{cases} \tag{5}$$

for $\phi_1, \phi_2$ smooth enough. The first term of the velocity, i.e., $\nabla^{\perp}\phi_1(x,y)$ is analogous to the vortexes defined in (12) and it is divergence-free, while the second term and the source terms balance each other. We will consider the smooth steady vortex (12) for the first part of $\underline{u}$, while we will use $\phi_2(x,y) := \frac{1}{100} e^{-100||\underline{x}-\underline{x_0}||_2^2}$, with $\underline{x_0} = (0.65, 0.39)^T$.

# Source term

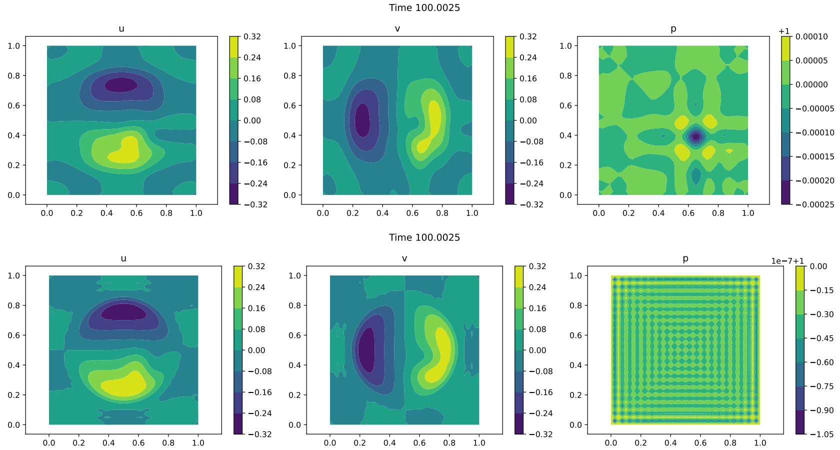



Figure: Simulation of vortex with source term at time $T = 100$ with $\mathbb{P}^1$ elements and $40 \times 40$ cells. SUPG scheme (top) and SUPG–GF scheme (bottom)

# Source term

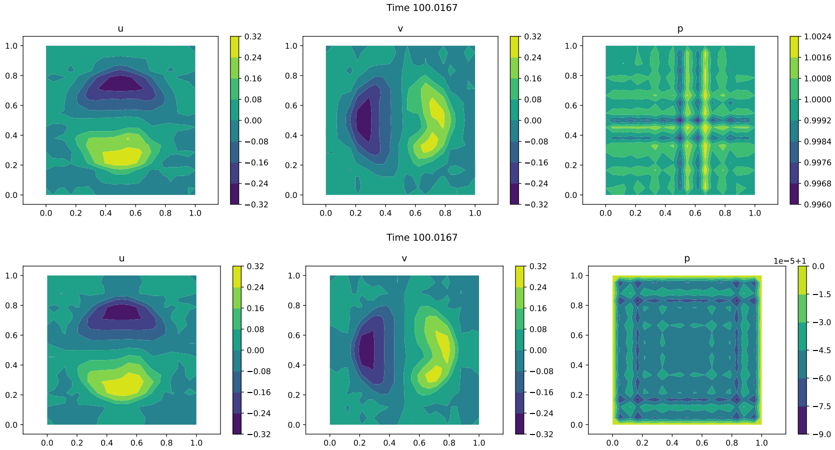
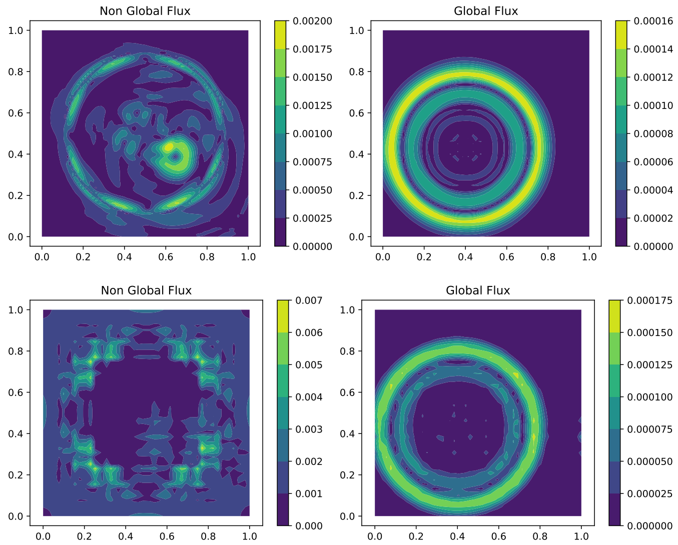



Figure: Simulation of vortex with source term at time $T = 100$ with $\mathbb{P}^3$ elements and $6 \times 6$ cells. SUPG scheme (top) and SUPG–GF scheme (bottom)

# Source term




## Vortex with Source

- Perturbation($\varepsilon = 10^{-3}$) test with source term.
- Plot of $\|\underline{u}_{eq} - \underline{u}_p\|$
- Top $\mathbb{P}^1$ with $80 \times 80$ cells
- Bottom $\mathbb{P}^3$ with 13 cells

# Stommel Gyre

## Stommel Gyre

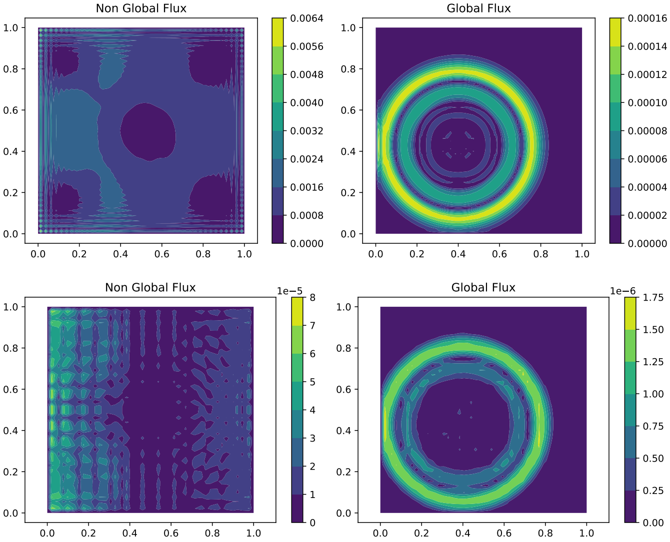
$$\partial_t p = -\operatorname{div}\mathbf{u}$$
$$\partial_t \mathbf{u} = -\operatorname{grad} p + \phi \mathbf{u}^\perp - R\mathbf{u} + \boldsymbol{\tau}$$

## Parameters

- $-R\mathbf{u}$ friction
- $\boldsymbol{\tau}$ wind forcing
- linearized shallow water equations, with a reference depth $h_0 = 1$, and with the gravity acceleration $g = 1$
- $R$ is constant
- $\boldsymbol{\tau} = (-F\cos(\pi y/b), 0)$
- $F$ constant
- well known steady solution due to Henry Stommel[a]

[a] H. Stommel, The westwards intensification of wind-driven ocean currents, Trans.Amer.Geophys.Union 29(2), 1948

# Stommel Gyre




## Perturbation test for SG

- Plot of $\|\underline{u}_{eq} - \underline{u}_p\|$
- Top $\mathbb{P}^1$ with $80 \times 80$ cells $\varepsilon = 10^{-3}$
- Bottom $\mathbb{P}^3$ with 13 cells $\varepsilon = 10^{-5}$

# 2D Riemann Problem

## 2D Riemann Problem

- Center $\underline{x}_0 = (0.5, 0.5)$
- Domain $\Omega = [0,1]^2$
- ICs

$$u(\underline{x}) = \begin{cases} 1, & \text{if } x > 0.5 \text{ and } y > 0.5, \\ 0, & \text{else,} \end{cases} \qquad v(\underline{x}) = 0, \qquad p(\underline{x}) = 0.$$

- The perpendicular component $v$ has a logarithmic singularity in the center of the RP for all $t > 0$:

$$v(x,y,t) = \frac{1}{2\pi}\mathcal{L}\left( \frac{\sqrt{(x-x_0)^2 + (y-y_0)^2}}{ct} \right),$$

$$\mathcal{L}(s) := \log\left( \frac{1+\sqrt{1-s^2}}{s} \right) = -\log\left(\frac{s}{2}\right) - \frac{s^2}{4} + \mathcal{O}(s^4).$$

# 2D Riemann Problem

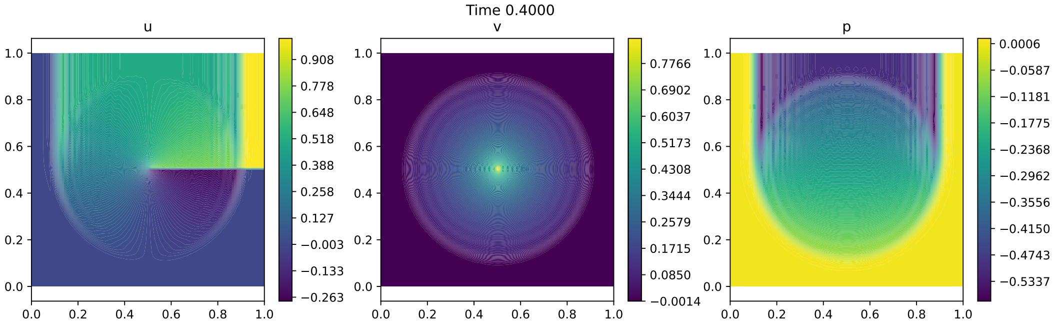



Figure: Riemann Problem. Simulation at time $T = 0.4$ with $\mathbb{P}^2$ elements and $50 \times 50$ cells with SUPG–GF scheme

# 2D Riemann Problem

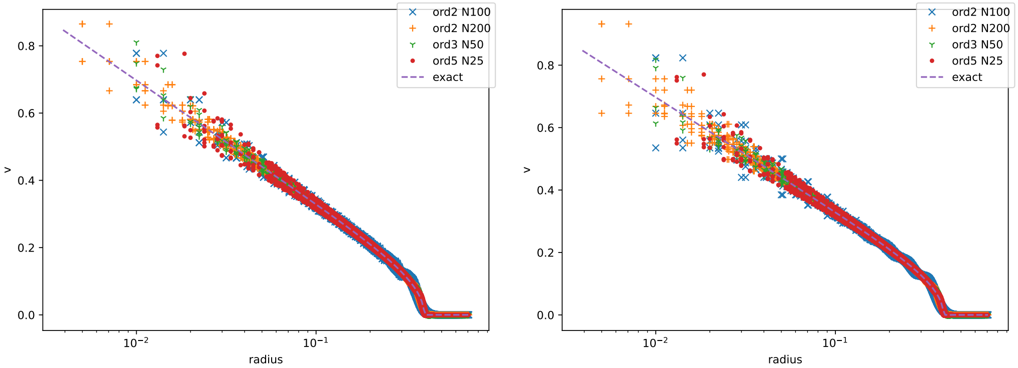



Figure: Riemann Problem. Distribution of the solution $v$ for different elements and meshes. Left SUPG scheme, right SUPG–GF scheme

# Involution

**Analytical involution**

$$\nabla \times \partial_t \underline{v} = \nabla \times (\nabla p) = 0$$

# Involution

## Analytical involution

$$\nabla \times \partial_t \underline{v} = \nabla \times (\nabla p) = 0$$

## Discrete involution

- If linear method of line $\partial_t Q = F(Q) \Longrightarrow Q^{n+1} = M(F, Q^n)$
- Operator $E$ such that $E \cdot (M(F,Q)) = M(E \cdot F, Q) = 0$ for all $Q$

# Involution

## Analytical involution

$$\nabla \times \partial_t \underline{v} = \nabla \times (\nabla p) = 0$$

## Discrete involution

- If linear method of line $\partial_t Q = F(Q) \implies Q^{n+1} = M(F, Q^n)$
- Operator $E$ such that $E \cdot (M(F,Q)) = M(E \cdot F, Q) = 0$ for all $Q$

## 2D SUPG $\mathbb{Q}^1$ involution operator

$$E := \begin{pmatrix} D_x\Big((D_y)^2 M_x - \alpha^2 \Delta y D_y^y \Big(\Delta y D_y^y M_x + \Delta x D_x^x M_y\Big)\Big) \\ D_y\Big(-(D_x)^2 M_y + \alpha^2 \Delta x D_x^x \Big(\Delta y D_y^y M_x + \Delta x D_x^x M_y\Big)\Big) \\ \alpha\left(-\Delta x D_x^x (D_y)^2 M_x + (D_x)^2 \Delta y D_y^y M_y\right) \end{pmatrix}.$$

# Involution

## Analytical involution

$$\nabla \times \partial_t \underline{v} = \nabla \times (\nabla p) = 0$$

## Discrete involution

- If linear method of line $\partial_t Q = F(Q) \implies Q^{n+1} = M(F, Q^n)$
- Operator $E$ such that $E \cdot (M(F,Q)) = M(E \cdot F, Q) = 0$ for all $Q$

## 2D SUPG $\mathbb{Q}^1$ involution operator

$$E := \begin{pmatrix} D_x\left( (D_y)^2 M_x - \alpha^2 \Delta y D_y^y \left( \Delta y D_y^y M_x + \Delta x D_x^x M_y \right)\right) \\ D_y\left( -(D_x)^2 M_y + \alpha^2 \Delta x D_x^x \left( \Delta y D_y^y M_x + \Delta x D_x^x M_y \right)\right) \\ \alpha \left( -\Delta x D_x^x (D_y)^2 M_x + (D_x)^2 \Delta y D_y^y M_y \right) \end{pmatrix}.$$

## 2D SUPG $\mathbb{Q}^p$ involution operator

Not feasible