

# Large scale Finite Element solvers for the large eddy simulation of incompressible turbulent flows

*author:*

Oriol Colomés Gené

*supervisor:*

Santiago Badia

Departament d'Enginyeria Civil i Ambiental

March 31 2016



**Escola de Camins**

Escola Tècnica Superior d'Enginyeria de Camins, Canals i Ports  
UPC BARCELONATECH

## 1. Motivation

## 2. Residual-based VMS

## 3. Mixed FE VMS

## 4. Segregated Runge-Kutta

## 5. Segregated VMS

## 6. Conclusions

## 1. Motivation

## 2. Residual-based VMS

## 3. Mixed FE VMS

## 4. Segregated Runge-Kutta

## 5. Segregated VMS

## 6. Conclusions

# Motivation

## Thesis motivation

# Motivation

## Thesis motivation

- Turbulence is a very challenging problem:
  - Many industrial applications
  - Understanding the phenomena (physics)

# Motivation

## Thesis motivation

- Turbulence is a very challenging problem:
  - Many industrial applications
  - Understanding the phenomena (physics)
- Computational Fluid Dynamics (CFD) to simulate such flows

# Motivation

## Thesis motivation

- Turbulence is a very challenging problem:
  - Many industrial applications
  - Understanding the phenomena (physics)
- Computational Fluid Dynamics (CFD) to simulate such flows
- Take advantage of increasing computational capacity  
(High-performance Computing (HPC))

# Motivation

## Thesis motivation

- Turbulence is a very challenging problem:
  - Many industrial applications
  - Understanding the phenomena (physics)
- Computational Fluid Dynamics (CFD) to simulate such flows
- Take advantage of increasing computational capacity  
(High-performance Computing (HPC))
- In a Finite Element (FE) framework

# Objective

## Thesis goal

Highly scalable Finite Element (FE) framework for Large Eddy Simulations (LES) of incompressible turbulent flows

# Objective

## Thesis goal

Highly scalable Finite Element (FE) framework for Large Eddy Simulations (LES) of incompressible turbulent flows

## How to get there?

# Objective

## Thesis goal

Highly scalable Finite Element (FE) framework for Large Eddy Simulations (LES) of incompressible turbulent flows

## How to get there?

1. Variational MultiScale (VMS) methods as LES models.

# Objective

## Thesis goal

Highly scalable Finite Element (FE) framework for Large Eddy Simulations (LES) of incompressible turbulent flows

## How to get there?

1. Variational MultiScale (VMS) methods as LES models.
2. Time integration schemes with **velocity-pressure segregation**.

# Objective

## Thesis goal

Highly scalable Finite Element (FE) framework for Large Eddy Simulations (LES) of incompressible turbulent flows

## How to get there?

1. Variational MultiScale (VMS) methods as LES models.
2. Time integration schemes with velocity-pressure segregation.
3. Highly scalable algorithms based on **Domain Decomposition (DD)** and **block preconditioners**.

# Objective

## Step by step...

- Residual-based VMS as LES models.

# Objective

## Step by step...

- Residual-based VMS as LES models.
- Mixed FE formulations LES.

# Objective

## Step by step...

- Residual-based VMS as LES models.
- Mixed FE formulations LES.
- High-order FE methods.

# Objective

## Step by step...

- Residual-based VMS as LES models.
- Mixed FE formulations LES.
- High-order FE methods.
- High-order time integration schemes.

# Objective

## Step by step...

- Residual-based VMS as LES models.
- Mixed FE formulations LES.
- High-order FE methods.
- High-order time integration schemes.
- Adaptive time stepping techniques.

# Objective

## Step by step...

- Residual-based VMS as LES models.
- Mixed FE formulations LES.
- High-order FE methods.
- High-order time integration schemes.
- Adaptive time stepping techniques.
- Velocity-pressure segregation.

# Objective

## Step by step...

- Residual-based VMS as LES models.
- Mixed FE formulations LES.
- High-order FE methods.
- High-order time integration schemes.
- Adaptive time stepping techniques.
- Velocity-pressure segregation.
- Scalable solvers.

# Objective

## Step by step...

- Residual-based VMS as LES models.
- Mixed FE formulations LES.
- High-order FE methods.
- High-order time integration schemes.
- Adaptive time stepping techniques.
- Velocity-pressure segregation.
- Scalable solvers.
- Application.

# Objective

## Step by step...

- Residual-based VMS as LES models.
- Mixed FE formulations LES.
- High-order FE methods.
- High-order time integration schemes.
- Adaptive time stepping techniques.
- Velocity-pressure segregation.
- Scalable solvers.
- Application.

## 1. Motivation

Rb-VMS

## 2. Residual-based VMS

Formulation

Numerical experiments

Conclusions

## 3. Mixed FE VMS

## 4. Segregated Runge-Kutta

## 5. Segregated VMS

## 6. Conclusions

# Implicit LES

**ILES:** only numerical dissipation (for stabilization) acts as turbulent model

- Not based on filtering of the Navier-Stokes equations
- Rely on numerical artifacts, no modification at the continuous level

# Incomp. Navier Stokes equations

Find  $\mathbf{u}$  and  $p$  defined in  $\Omega$

$$\partial_t \mathbf{u} + \mathbf{u} \cdot \nabla \mathbf{u} + \nabla p - \nu \nabla^2 \mathbf{u} = \mathbf{f}$$

$$\nabla \cdot \mathbf{u} = 0$$

with appropriate boundary conditions on  $\Gamma$ .

# Incomp. Navier Stokes equations

Find  $\mathbf{u}$  and  $p$  defined in  $\Omega$

$$\partial_t \mathbf{u} + \mathbf{u} \cdot \nabla \mathbf{u} + \nabla p - \nu \nabla^2 \mathbf{u} = \mathbf{f}$$

$$\nabla \cdot \mathbf{u} = 0$$

with appropriate boundary conditions on  $\Gamma$ .

The weak problem is:  $\forall \mathbf{v} \in \mathcal{V}_0$  and  $\forall q \in \mathcal{Q}_0$ , find  $\mathbf{u} \in \mathcal{V}$  and  $p \in \mathcal{Q}$  such that

$$\begin{aligned} (\mathbf{v}, \partial_t \mathbf{u})_\Omega + (\nabla \mathbf{v}, \nu \nabla \mathbf{u})_\Omega + b(\mathbf{u}, \mathbf{u}, \mathbf{v}) - (\nabla \cdot \mathbf{v}, p)_\Omega &= \langle \mathbf{v}, \mathbf{f} \rangle_\Omega \\ (q, \nabla \cdot \mathbf{u})_\Omega &= 0 \end{aligned}$$

# Incomp. Navier Stokes equations

Find  $\mathbf{u}$  and  $p$  defined in  $\Omega$

$$\partial_t \mathbf{u} + \mathbf{u} \cdot \nabla \mathbf{u} + \nabla p - \nu \nabla^2 \mathbf{u} = \mathbf{f}$$

$$\nabla \cdot \mathbf{u} = 0$$

with appropriate boundary conditions on  $\Gamma$ .

The weak problem is:  $\forall \mathbf{v} \in \mathcal{V}_0$  and  $\forall q \in \mathcal{Q}_0$ , find  $\mathbf{u} \in \mathcal{V}$  and  $p \in \mathcal{Q}$  such that

$$\begin{aligned} (\mathbf{v}, \partial_t \mathbf{u})_\Omega + (\nabla \mathbf{v}, \nu \nabla \mathbf{u})_\Omega + b(\mathbf{u}, \mathbf{u}, \mathbf{v}) - (\nabla \cdot \mathbf{v}, p)_\Omega &= \langle \mathbf{v}, \mathbf{f} \rangle_\Omega \\ (q, \nabla \cdot \mathbf{u})_\Omega &= 0 \end{aligned}$$

where

$$b(\mathbf{a}, \mathbf{u}, \mathbf{v}) = \langle \mathbf{v}, \mathbf{a} \cdot \nabla \mathbf{u} \rangle_\Omega$$

# Incomp. Navier Stokes equations

Find  $\mathbf{u}$  and  $p$  defined in  $\Omega$

$$\partial_t \mathbf{u} + \mathbf{u} \cdot \nabla \mathbf{u} + \nabla p - \nu \nabla^2 \mathbf{u} = \mathbf{f}$$

$$\nabla \cdot \mathbf{u} = 0$$

with appropriate boundary conditions on  $\Gamma$ .

The weak problem is:  $\forall \mathbf{v} \in \mathcal{V}_0$  and  $\forall q \in \mathcal{Q}_0$ , find  $\mathbf{u} \in \mathcal{V}$  and  $p \in \mathcal{Q}$  such that

$$\begin{aligned} (\mathbf{v}, \partial_t \mathbf{u})_\Omega + (\nabla \mathbf{v}, \nu \nabla \mathbf{u})_\Omega + b(\mathbf{u}, \mathbf{u}, \mathbf{v}) - (\nabla \cdot \mathbf{v}, p)_\Omega &= \langle \mathbf{v}, \mathbf{f} \rangle_\Omega \\ (q, \nabla \cdot \mathbf{u})_\Omega &= 0 \end{aligned}$$

where

$$b(\mathbf{a}, \mathbf{u}, \mathbf{v}) = \frac{1}{2} \langle \mathbf{v}, \mathbf{a} \cdot \nabla \mathbf{u} \rangle_\Omega - \frac{1}{2} \langle \mathbf{a} \cdot \nabla \mathbf{v}, \mathbf{u} \rangle_\Omega + \frac{1}{2} \langle \mathbf{v}, \mathbf{n} \cdot \mathbf{a} \mathbf{u} \rangle_\Gamma$$

# VMS decomposition (Hughes 1995)

A decomposition of spaces  $\mathcal{V}$  and  $\mathcal{Q}$  given by

$$\mathcal{V} = \mathcal{V}_h \oplus \tilde{\mathcal{V}}, \quad \mathcal{Q} = \mathcal{Q}_h \oplus \tilde{\mathcal{Q}}$$

## VMS decomposition (Hughes 1995)

A decomposition of spaces  $\mathcal{V}$  and  $\mathcal{Q}$  given by

$$\mathcal{V} = \mathcal{V}_h \oplus \tilde{\mathcal{V}}, \quad \mathcal{Q} = \mathcal{Q}_h \oplus \tilde{\mathcal{Q}}$$

is applied to the function and test spaces

$$\mathbf{u} = \mathbf{u}_h + \tilde{\mathbf{u}}, \quad p = p_h + \tilde{p}$$

$$\mathbf{v} = \mathbf{v}_h + \tilde{\mathbf{v}}, \quad q = q_h + \tilde{q}$$

## VMS decomposition (Hughes 1995)

A decomposition of spaces  $\mathcal{V}$  and  $\mathcal{Q}$  given by

$$\mathcal{V} = \mathcal{V}_h \oplus \tilde{\mathcal{V}}, \quad \mathcal{Q} = \mathcal{Q}_h \oplus \tilde{\mathcal{Q}}$$

is applied to the function and test spaces

$$\begin{aligned}\mathbf{u} &= \mathbf{u}_h + \tilde{\mathbf{u}}, & p &= p_h + \tilde{p} \\ \mathbf{v} &= \mathbf{v}_h + \tilde{\mathbf{v}}, & q &= q_h + \tilde{q}\end{aligned}$$

We keep all the (eight) contributions from the splitting of the convective term

$$\mathbf{u} \cdot \nabla \mathbf{u} = \mathbf{u}_h \cdot \nabla \mathbf{u}_h + \tilde{\mathbf{u}} \cdot \nabla \mathbf{u}_h + \mathbf{u}_h \cdot \nabla \tilde{\mathbf{u}} + \tilde{\mathbf{u}} \cdot \nabla \tilde{\mathbf{u}}$$

## VMS decomposition (Hughes 1995)

A decomposition of spaces  $\mathcal{V}$  and  $\mathcal{Q}$  given by

$$\mathcal{V} = \mathcal{V}_h \oplus \tilde{\mathcal{V}}, \quad \mathcal{Q} = \mathcal{Q}_h \oplus \tilde{\mathcal{Q}}$$

is applied to the function and test spaces

$$\begin{aligned}\mathbf{u} &= \mathbf{u}_h + \tilde{\mathbf{u}}, & p &= p_h + \tilde{p} \\ \mathbf{v} &= \mathbf{v}_h + \tilde{\mathbf{v}}, & q &= q_h + \tilde{q}\end{aligned}$$

We keep all the (eight) contributions from the splitting of the convective term

$$\mathbf{u} \cdot \nabla \mathbf{u} = \mathbf{u}_h \cdot \nabla \mathbf{u}_h + \tilde{\mathbf{u}} \cdot \nabla \mathbf{u}_h + \mathbf{u}_h \cdot \nabla \tilde{\mathbf{u}} + \tilde{\mathbf{u}} \cdot \nabla \tilde{\mathbf{u}}$$

and all the (four) contributions from the temporal term

$$\partial_t \mathbf{u} = \partial_t \mathbf{u}_h + \partial_t \tilde{\mathbf{u}}$$

# Semidiscrete problem

## FEM equations

$$B((\mathbf{u}_h, p_h); (\tilde{\mathbf{u}}, \tilde{p}) ; (\mathbf{v}_h, q_h)) = L(\mathbf{v}_h, q_h)$$

## SGS equations

$$B((\tilde{\mathbf{u}}, \tilde{p}) ; (\mathbf{u}_h, p_h) ; (\tilde{\mathbf{v}}, \tilde{q})) = L(\tilde{\mathbf{v}}, \tilde{q})$$

# Semidiscrete problem

## FEM equations

$$\begin{aligned}
 & (\mathbf{v}_h, \partial_t \mathbf{u}_h)_\Omega + b(\mathbf{a}, \mathbf{u}_h, \mathbf{v}_h) + (\nabla \mathbf{v}_h, \nu \nabla \mathbf{u}_h)_\Omega - (\nabla \cdot \mathbf{v}_h, p_h)_\Omega \\
 & + (\mathbf{v}_h, \partial_t \tilde{\mathbf{u}})_\Omega + (\mathcal{L}^* \mathbf{v}_h, \tilde{\mathbf{u}})_{\Omega^h} - (\nabla \cdot \mathbf{v}_h, \tilde{p})_{\Omega^h} = \langle \mathbf{v}_h, \mathbf{f} \rangle_\Omega \\
 & (q_h, \nabla \cdot \mathbf{u}_h)_\Omega - (\nabla q_h, \tilde{\mathbf{u}})_{\Omega^h} = 0
 \end{aligned}$$

## SGS equations

$$\partial_t \tilde{\mathbf{u}} + \tau_m^{-1} \tilde{\mathbf{u}} = \mathcal{P} \mathbf{R}_m$$

$$\tau_c^{-1} \tilde{p} = \mathcal{P} R_c$$

$$\tau_m = \left( \frac{c_1 \nu}{h^2} + \frac{c_2 |\mathbf{a}|}{h} \right)^{-1}, \quad \tau_c = \frac{h^2}{c_1 \tau_m}$$

$$\mathbf{a} = \mathbf{u}_h + \tilde{\mathbf{u}}$$

# Semidiscrete problem

## FEM equations

$$\begin{aligned}
 & (\mathbf{v}_h, \partial_t \mathbf{u}_h)_\Omega + b(\mathbf{a}, \mathbf{u}_h, \mathbf{v}_h) + (\nabla \mathbf{v}_h, \nu \nabla \mathbf{u}_h)_\Omega - (\nabla \cdot \mathbf{v}_h, p_h)_\Omega \\
 & + (\mathbf{v}_h, \partial_t \tilde{\mathbf{u}})_\Omega + (\mathcal{L}^* \mathbf{v}_h, \tilde{\mathbf{u}})_{\Omega^h} - (\nabla \cdot \mathbf{v}_h, \tilde{p})_{\Omega^h} = \langle \mathbf{v}_h, \mathbf{f} \rangle_\Omega \\
 & (q_h, \nabla \cdot \mathbf{u}_h)_\Omega - (\nabla q_h, \tilde{\mathbf{u}})_{\Omega^h} = 0
 \end{aligned}$$

## SGS equations

$$\partial_t \tilde{\mathbf{u}} + \tau_m^{-1} \tilde{\mathbf{u}} = \mathcal{P} \mathbf{R}_m$$

$$\tau_c^{-1} \tilde{p} = \mathcal{P} \mathbf{R}_c$$

$$\mathbf{R}_m := \mathbf{f} - \partial_t \mathbf{u}_h - \mathcal{L} \mathbf{u}_h - \nabla p_h, \quad \mathbf{R}_c := -\nabla \cdot \mathbf{u}_h$$

# Semidiscrete problem

## FEM equations

$$\begin{aligned}
 & (\mathbf{v}_h, \partial_t \mathbf{u}_h)_\Omega + b(\mathbf{a}, \mathbf{u}_h, \mathbf{v}_h) + (\nabla \mathbf{v}_h, \nu \nabla \mathbf{u}_h)_\Omega - (\nabla \cdot \mathbf{v}_h, p_h)_\Omega \\
 & + (\mathbf{v}_h, \partial_t \tilde{\mathbf{u}})_\Omega + (\mathcal{L}^* \mathbf{v}_h, \tilde{\mathbf{u}})_{\Omega^h} - (\nabla \cdot \mathbf{v}_h, \tilde{p})_{\Omega^h} = \langle \mathbf{v}_h, \mathbf{f} \rangle_\Omega \\
 & (q_h, \nabla \cdot \mathbf{u}_h)_\Omega - (\nabla q_h, \tilde{\mathbf{u}})_{\Omega^h} = 0
 \end{aligned}$$

## SGS equations

$$\partial_t \tilde{\mathbf{u}} + \tau_m^{-1} \tilde{\mathbf{u}} = \mathcal{P} \mathbf{R}_m$$

$$\tau_c^{-1} \tilde{p} = \mathcal{P} R_c$$

$$\mathcal{P} = I \quad (\text{ASGS}), \quad \mathcal{P} = P_h^\perp = I - P_h \quad (\text{OSS})$$

# Semidiscrete problem

## FEM equations

$$\begin{aligned}
 & (\mathbf{v}_h, \partial_t \mathbf{u}_h)_\Omega + b(\mathbf{a}, \mathbf{u}_h, \mathbf{v}_h) + (\nabla \mathbf{v}_h, \nu \nabla \mathbf{u}_h)_\Omega - (\nabla \cdot \mathbf{v}_h, p_h)_\Omega \\
 & + (\mathbf{v}_h, \partial_t \tilde{\mathbf{u}})_\Omega + (\mathcal{L}^* \mathbf{v}_h, \tilde{\mathbf{u}})_{\Omega^h} - (\nabla \cdot \mathbf{v}_h, \tilde{p})_{\Omega^h} = \langle \mathbf{v}_h, \mathbf{f} \rangle_\Omega \\
 & (q_h, \nabla \cdot \mathbf{u}_h)_\Omega - (\nabla q_h, \tilde{\mathbf{u}})_{\Omega^h} = 0
 \end{aligned}$$

## SGS equations

$$\partial_t \tilde{\mathbf{u}} + \tau_m^{-1} \tilde{\mathbf{u}} = \mathcal{P} \mathbf{R}_m \quad (\text{static/dynamic})$$

$$\tau_c^{-1} \tilde{p} = \mathcal{P} R_c$$

$$\mathcal{P} = I \quad (\text{ASGS}), \quad \mathcal{P} = P_h^\perp = I - P_h \quad (\text{OSS})$$

$$\mathbf{a} = \mathbf{u}_h + \tilde{\mathbf{u}} \quad (\text{linear/nonlinear})$$

# Numerical experiments

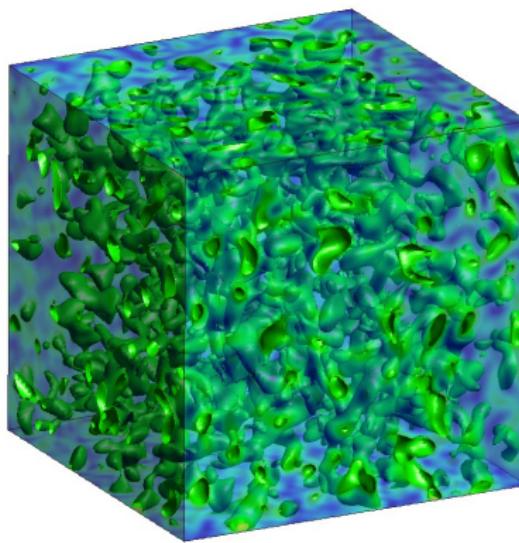
Three different turbulent benchmarks:

- Decaying of Homogeneous Isotropic Turbulence (DHIT).
- Taylor-Green Vortex (TGV) flow.
- Turbulent Channel Flow (TCF).

# DHIT Decay of Homogeneous Isotropic Turbulence

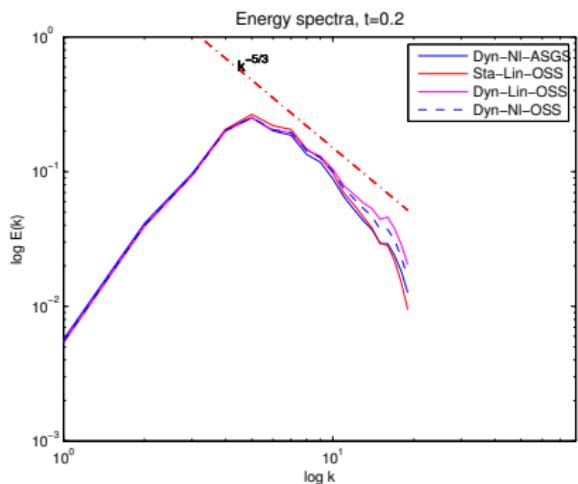
## Problem setting:

- Prescribed initial energy spectra corresponding to  $Re_\lambda = 952$ .
- Setting defined in AGARD database (Mansour & Wray 1993).
- A (very simple) time step adaptation technique is used.
- Different mesh discretizations ( $Q_1/Q_1$  and  $Q_2/Q_2$ ).

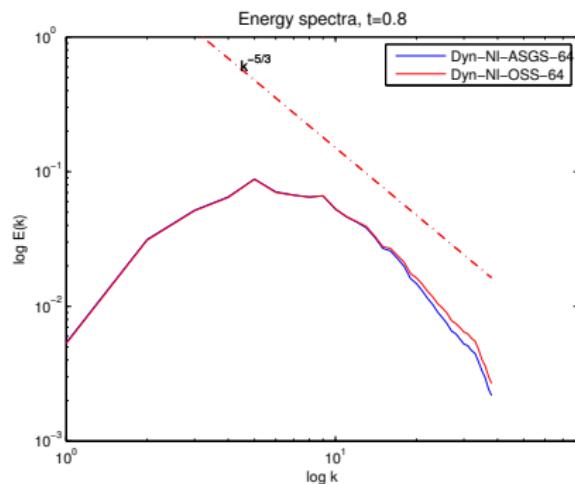


# DHIT Decay of Homogeneous Isotropic Turbulence

## Energy spectra (models):



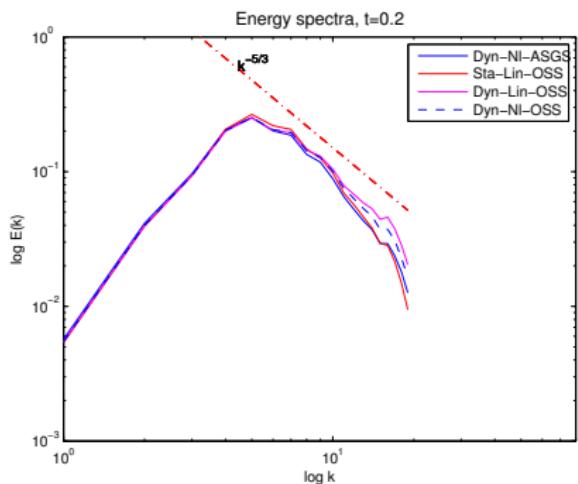
$32^3 - Q1, t = 0.2s$



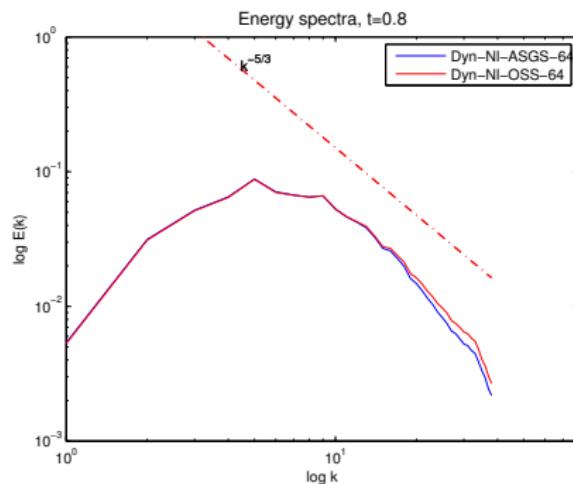
$64^3 - Q1, t = 0.8s$

# DHIT Decay of Homogeneous Isotropic Turbulence

## Energy spectra (models):



$32^3 - Q1, t = 0.2s$

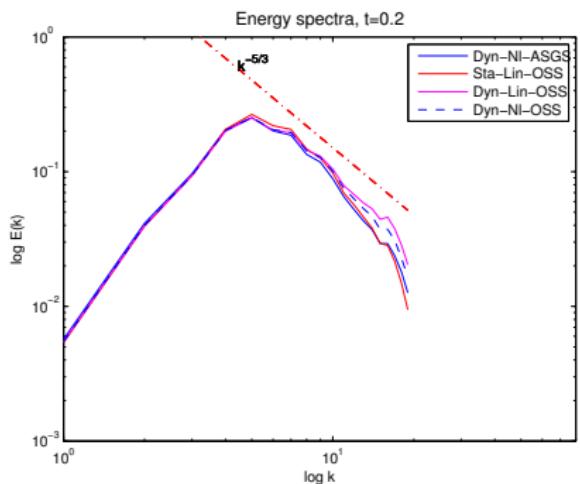


$64^3 - Q1, t = 0.8s$

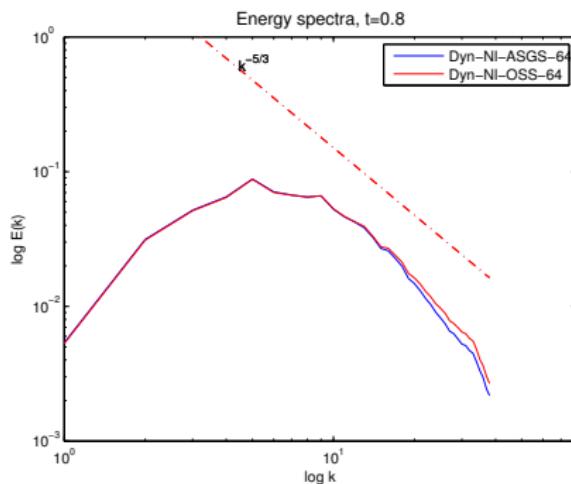
- Small differences between methods (physical sense).

# DHIT Decay of Homogeneous Isotropic Turbulence

## Energy spectra (models):



$32^3 - Q1, t = 0.2s$

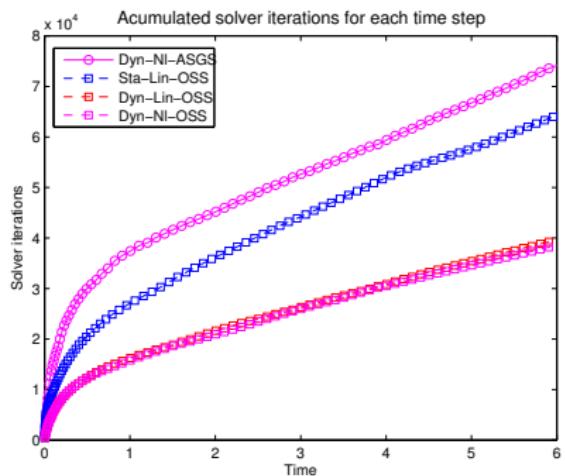


$64^3 - Q1, t = 0.8s$

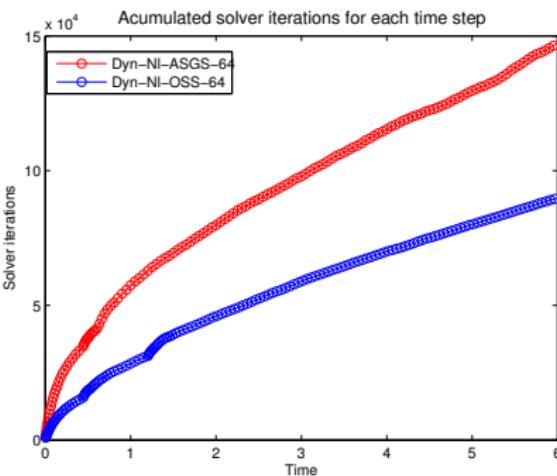
- Small differences between methods (physical sense).
- Even more similar when we refine the mesh.

# DHIT Decay of Homogeneous Isotropic Turbulence

## Computational cost (models):



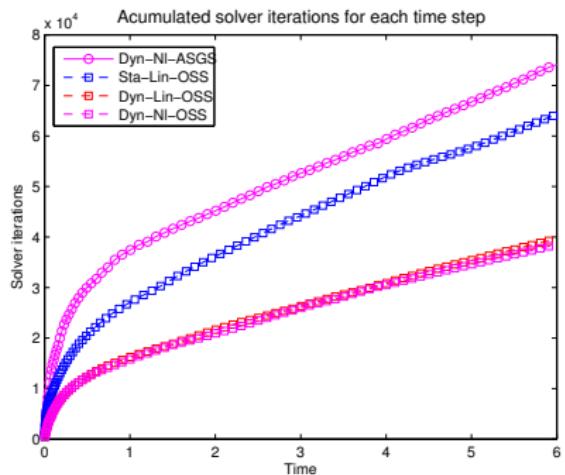
$32^3 - Q1$



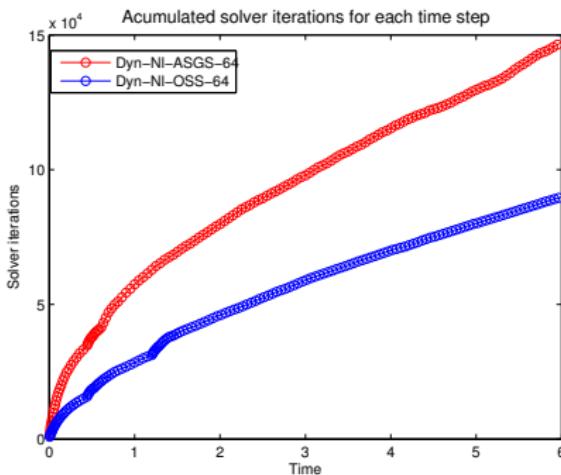
$64^3 - Q1$

# DHIT Decay of Homogeneous Isotropic Turbulence

## Computational cost (models):



$32^3 - Q1$

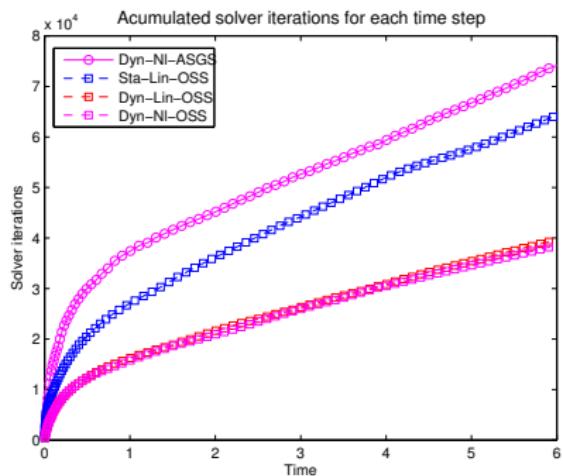


$64^3 - Q1$

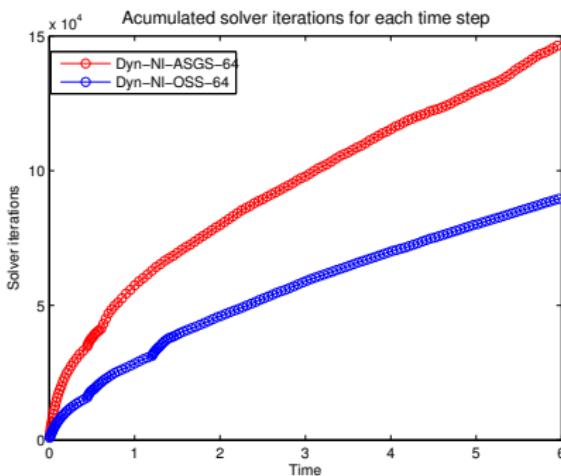
- Big differences between methods (computational sense).

# DHIT Decay of Homogeneous Isotropic Turbulence

## Computational cost (models):



$32^3 - Q1$

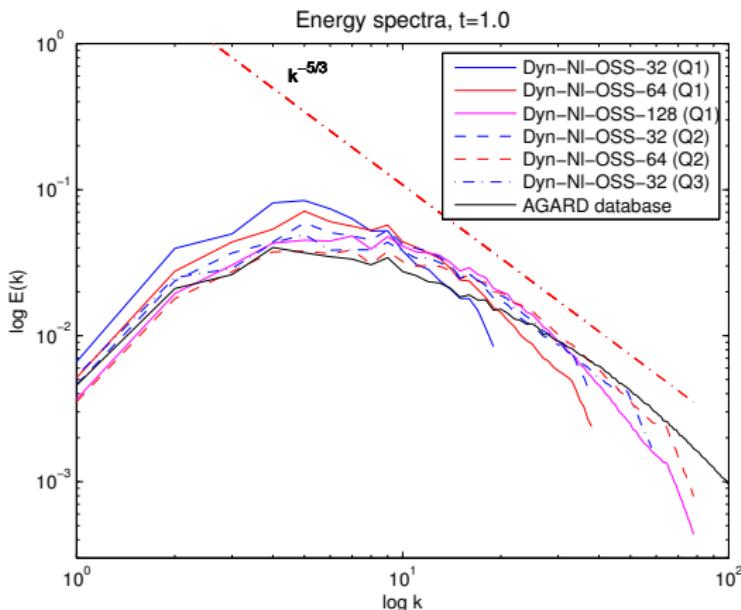


$64^3 - Q1$

- Big differences between methods (computational sense).
- **Dynamic** versions of **OSS** method are the most efficient.

# DHIT Decay of Homogeneous Isotropic Turbulence

## Energy spectra (refinement):

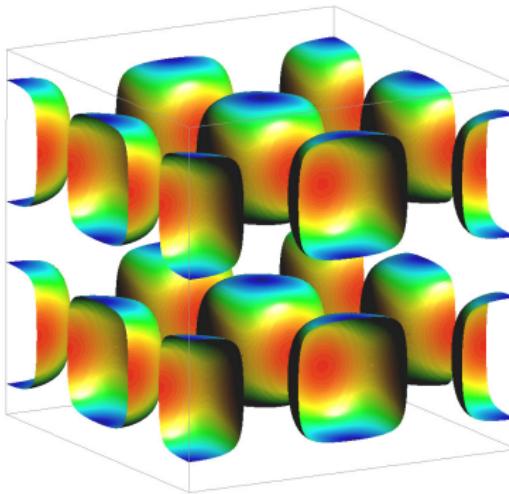


- Results become closer to the DNS when we refine the mesh.

# TGV Taylor-Green Vortex flow

## Problem setting:

- Prescribed initial condition.
- $Re = 1600$ .
- Different mesh discretizations ( $Q_1/Q_1$  and  $Q_2/Q_2$ ).

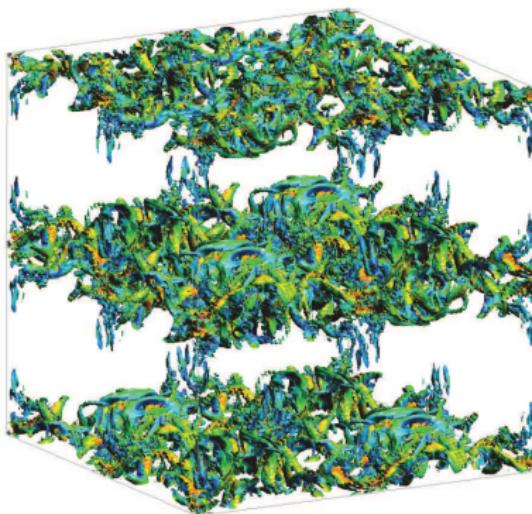


Initial vorticity isosurface  $|\omega| = 1$

# TGV Taylor-Green Vortex flow

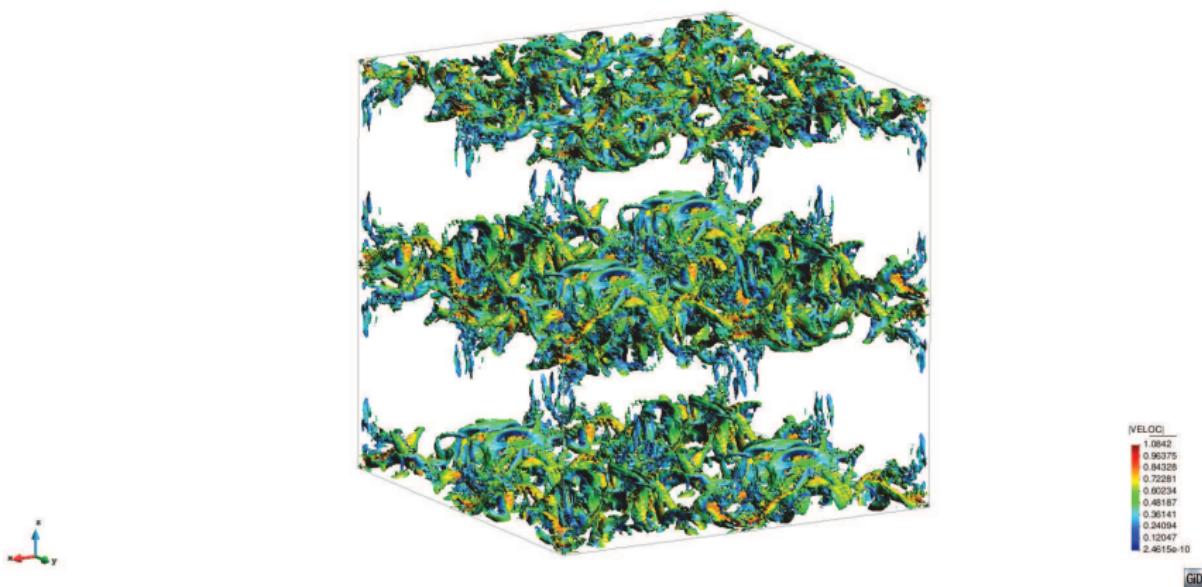
## Problem setting:

- Prescribed initial condition.
- $Re = 1600$ .
- Different mesh discretizations ( $Q_1/Q_1$  and  $Q_2/Q_2$ ).



Vorticity isosurfaces  $|\omega| = 9.0$

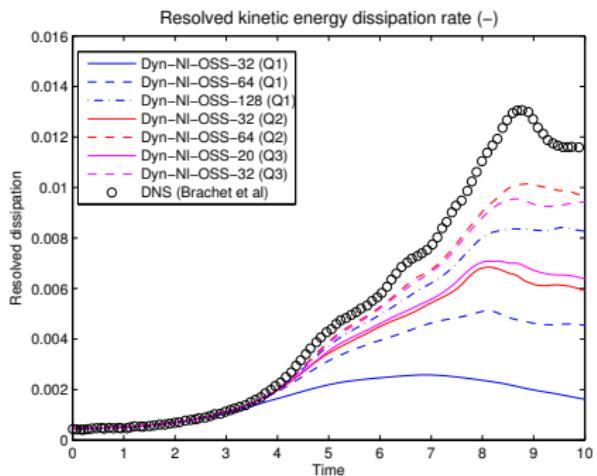
# TGV Taylor-Green Vortex flow



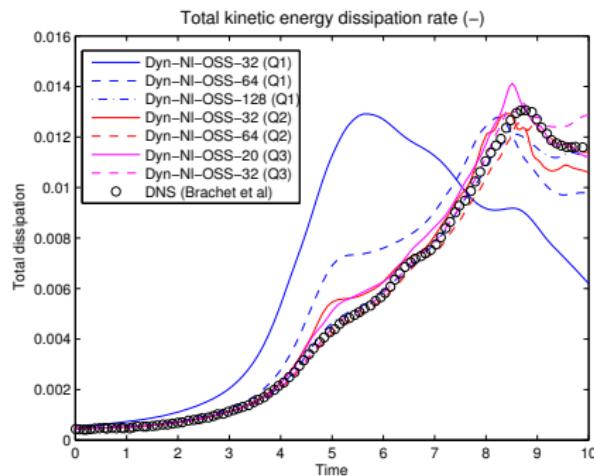
Velocity isosurface

# TGV Taylor-Green Vortex flow

## Energy dissipation rate (refinement):



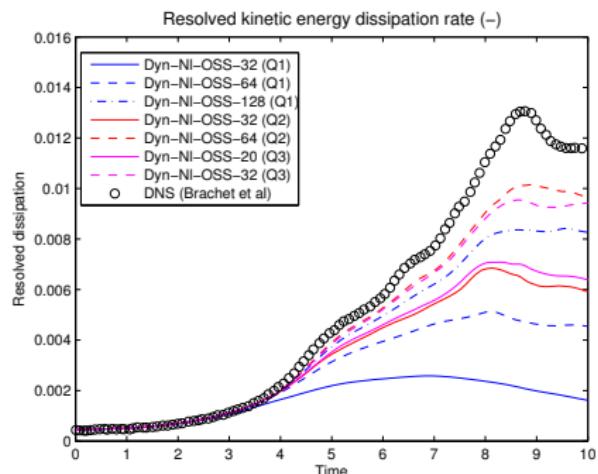
Resolved scales



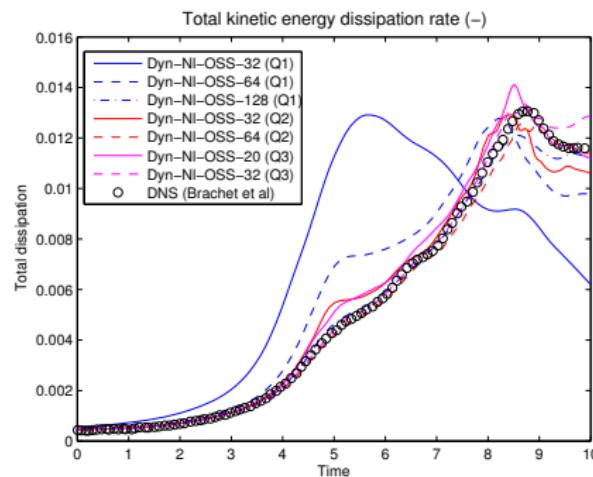
Total

# TGV Taylor-Green Vortex flow

## Energy dissipation rate (refinement):



Resolved scales

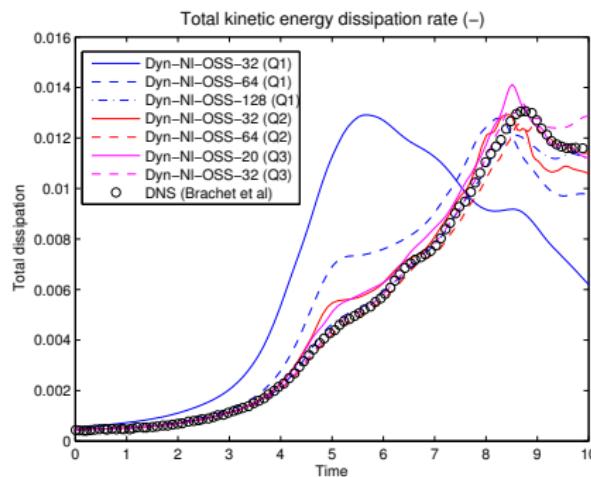
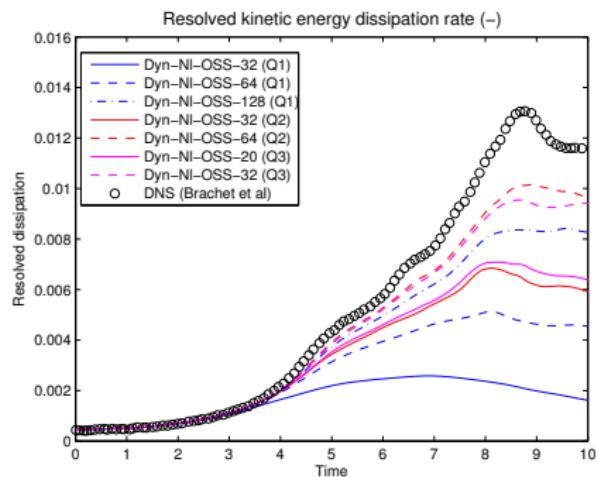


Total

- Good agreement with the DNS taking account the subscales.

# TGV Taylor-Green Vortex flow

## Energy dissipation rate (refinement):



Resolved scales

Total

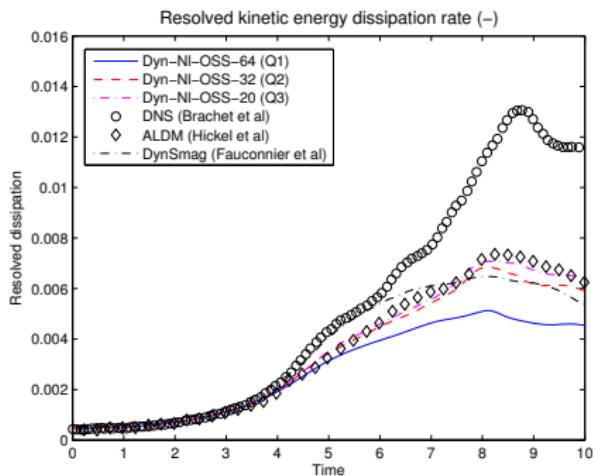
- Good agreement with the DNS taking account the subscales.
- More accurate results increasing the order of approximation.

# TGV Taylor-Green Vortex flow

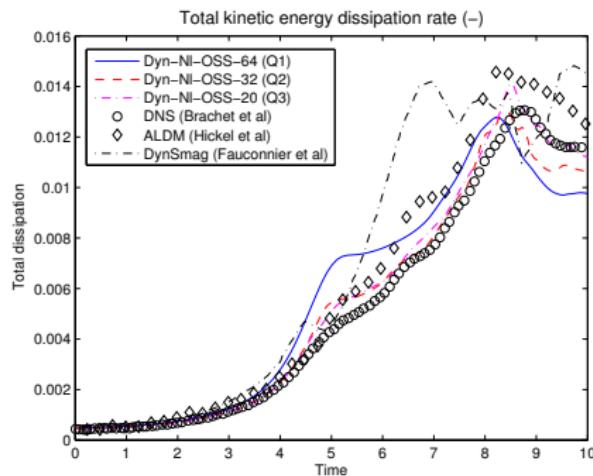
- All results until now are compared against **DNS**.
- Are our methods comparable with **LES** models?

# TGV Taylor-Green Vortex flow

## Energy dissipation rate (against LES model):



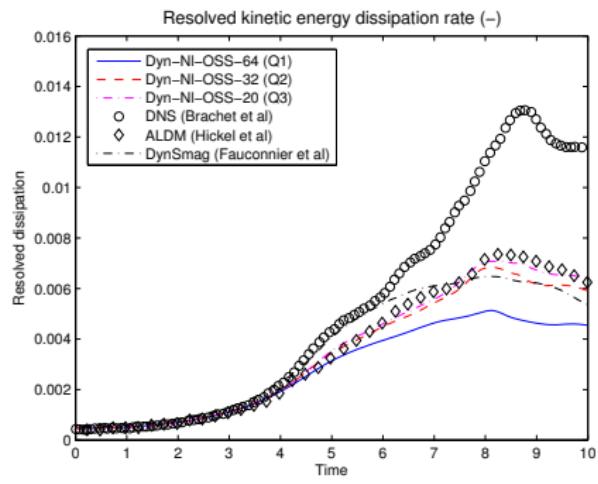
Resolved scales



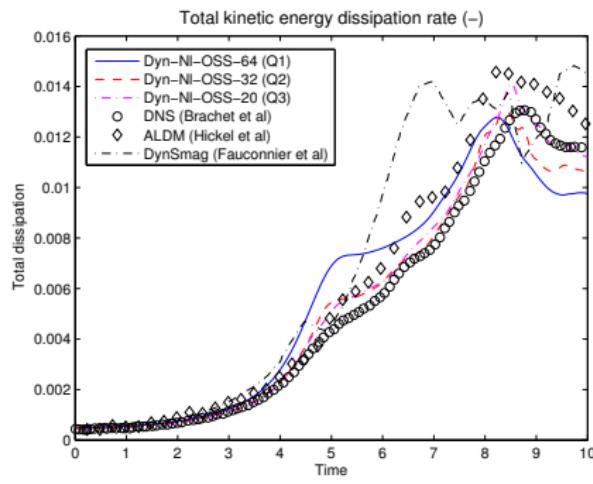
Total

# TGV Taylor-Green Vortex flow

## Energy dissipation rate (against LES model):



Resolved scales

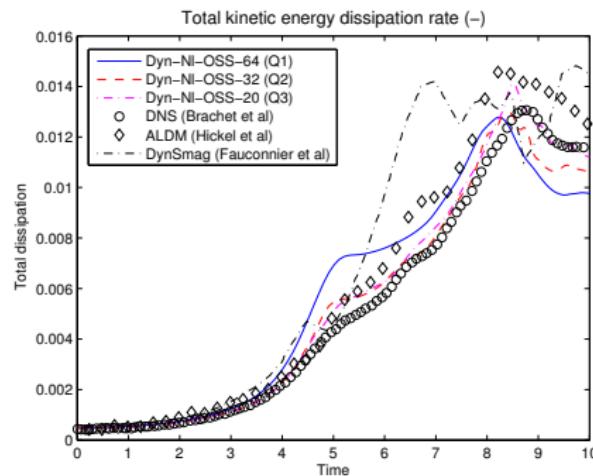
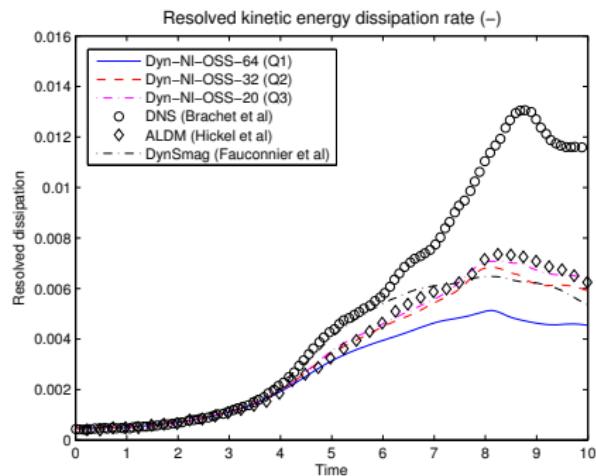


Total

- Good agreement with the LES models on resolved scales.

# TGV Taylor-Green Vortex flow

## Energy dissipation rate (against LES model):



Resolved scales

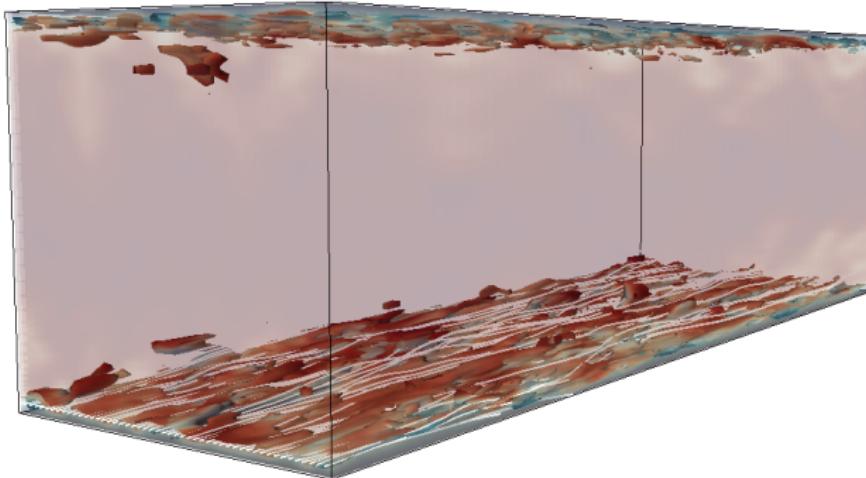
Total

- Good agreement with the LES models on resolved scales.
- Better results than LES models adding subscales counterpart.

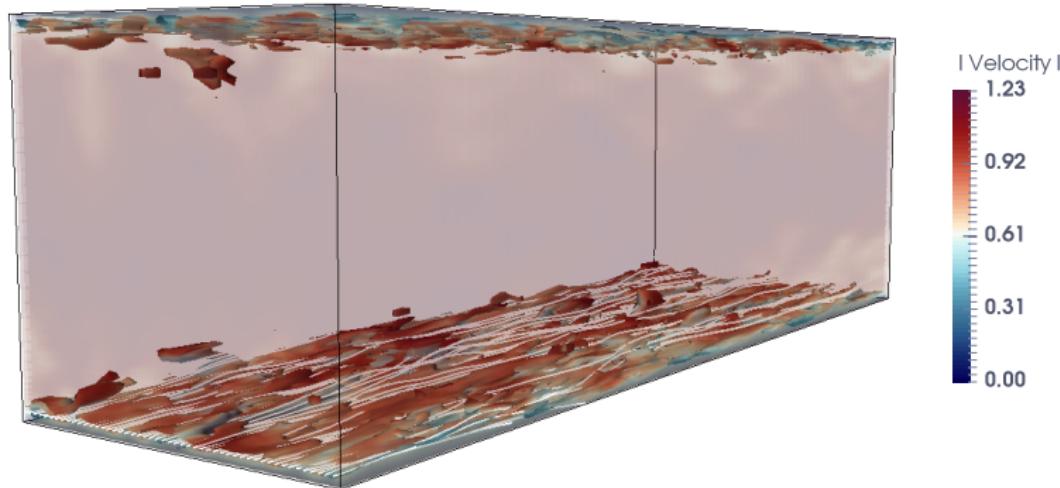
# TCF Turbulent Channel Flow

## Problem setting:

- Wall bounded flow.
- $Re_\tau = 180$  and  $Re_\tau = 395$ .
- Mesh resolution:  $32^3 - Q1$ .

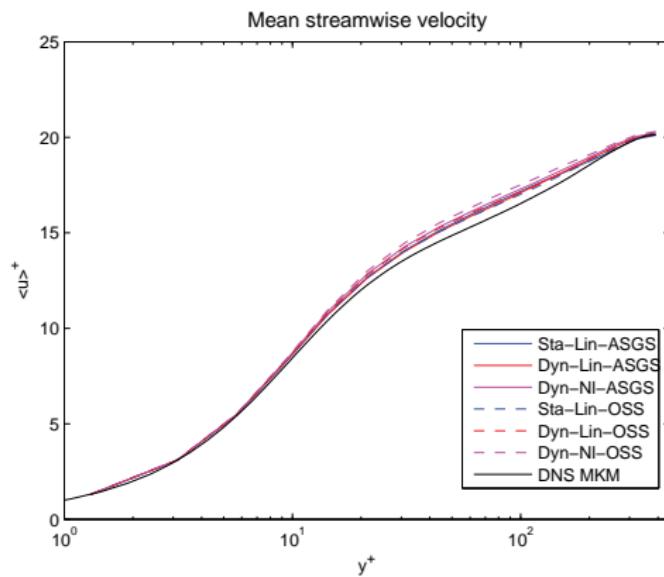


# TCF Turbulent Channel Flow



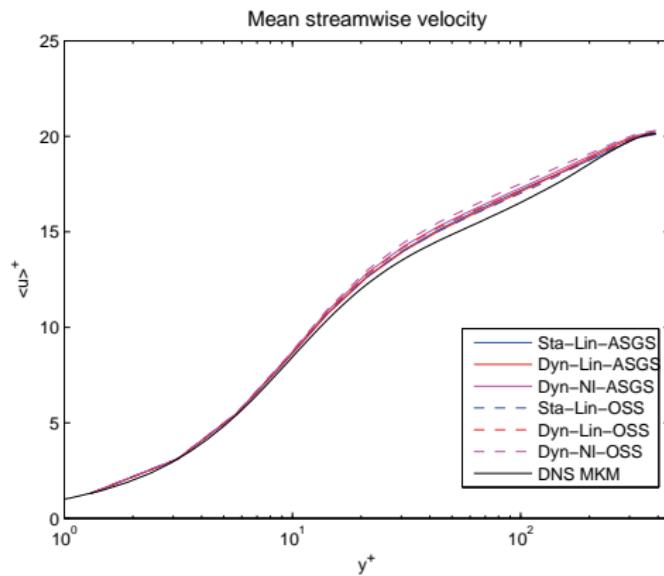
# TCF Turbulent Channel Flow

## Mean streamwise velocity (models):



# TCF Turbulent Channel Flow

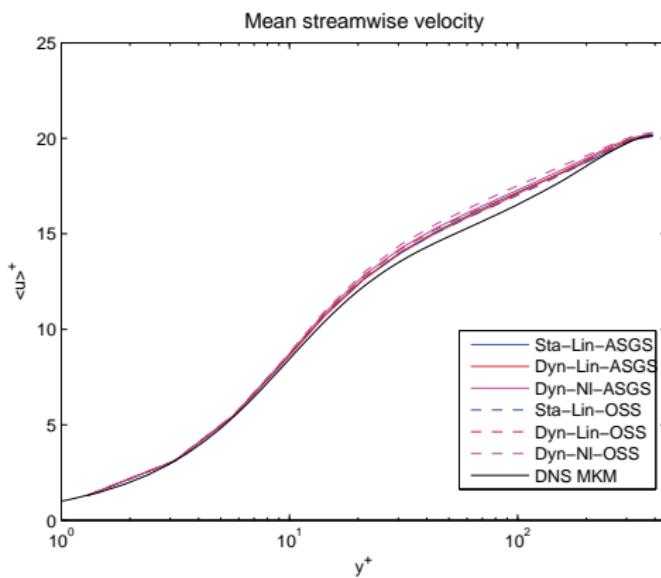
## Mean streamwise velocity (models):



- Small differences between methods.

# TCF Turbulent Channel Flow

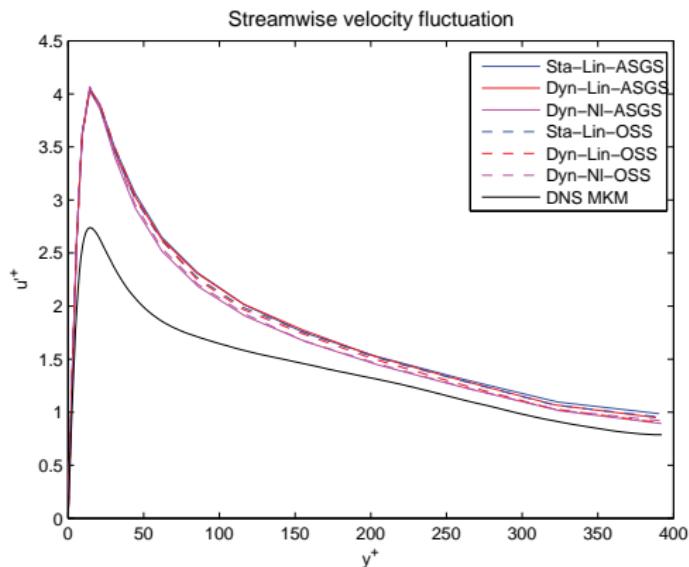
## Mean streamwise velocity (models):



- Small differences between methods.
- Very accurate results compared against the DNS.

# TCF Turbulent Channel Flow

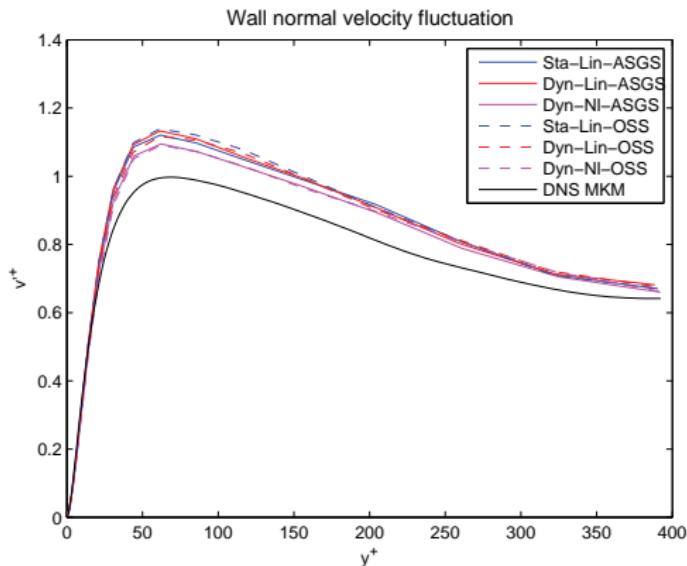
## Mean streamwise velocity (models):



- Small differences between methods.
- Very accurate results compared against the DNS.

# TCF Turbulent Channel Flow

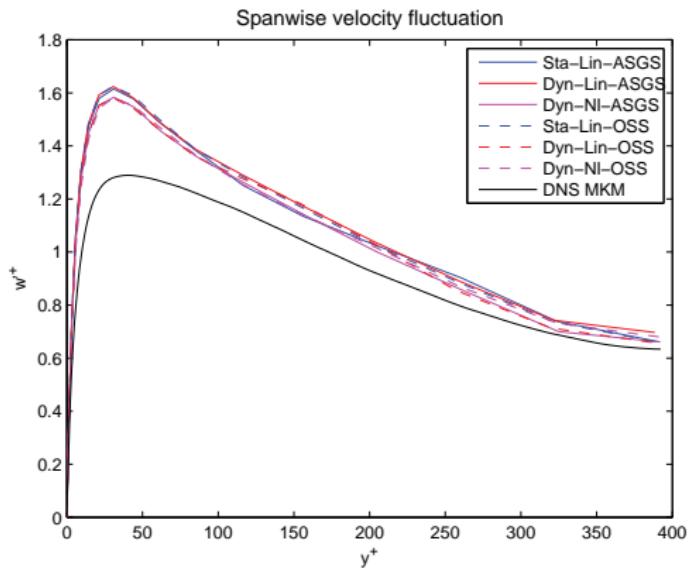
## Mean streamwise velocity (models):



- Small differences between methods.
- Very accurate results compared against the DNS.

# TCF Turbulent Channel Flow

## Mean streamwise velocity (models):



- Small differences between methods.
- Very accurate results compared against the DNS.

# RB-VMS Conclusions

- VMS formulations of NS equations can be used for the numerical simulation of turbulent flows.

# RB-VMS Conclusions

- VMS formulations of NS equations can be used for the numerical simulation of turbulent flows.
  - Our particular VMS modelling ingredients
    - Dynamic subscales
    - Nonlinear subscales
    - Orthgonal subscales
- seem to be important to reduce the computational cost.

# RB-VMS Conclusions

- VMS formulations of NS equations can be used for the numerical simulation of turbulent flows.
- Our particular VMS modelling ingredients
  - Dynamic subscales
  - Nonlinear subscales
  - Orthogonal subscalesseem to be important to reduce the computational cost.
- Among them dynamic and orthogonal subscales (linear or nonlinear) are the most effective.

# RB-VMS Conclusions

- VMS formulations of NS equations can be used for the numerical simulation of turbulent flows.
- Our particular VMS modelling ingredients
  - Dynamic subscales
  - Nonlinear subscales
  - Orthogonal subscalesseem to be important to reduce the computational cost.
- Among them dynamic and orthogonal subscales (linear or nonlinear) are the most effective.
- The skewsymmetric formulation is important to keep stability.

# RB-VMS Limitations

- ASGS:

# RB-VMS Limitations

- **ASGS:**

- Poorly matrix conditioning  $\Rightarrow$  **High** number of **solver** iterations.

# RB-VMS Limitations

- **ASGS:**

- Poorly matrix conditioning  $\Rightarrow$  **High** number of **solver** iterations.
- No explicit projections  $\Rightarrow$  **Low** number of **nonlinear** iterations.

# RB-VMS Limitations

- **ASGS:**

- Poorly matrix conditioning  $\Rightarrow$  **High** number of **solver** iterations.
- No explicit projections  $\Rightarrow$  **Low** number of **nonlinear** iterations.

- **OSS:**

# RB-VMS Limitations

- **ASGS:**

- Poorly matrix conditioning  $\Rightarrow$  **High** number of **solver** iterations.
- No explicit projections  $\Rightarrow$  **Low** number of **nonlinear** iterations.

- **OSS:**

- Better matrix conditioning  $\Rightarrow$  **Low** number of **solver** iterations.

# RB-VMS Limitations

- **ASGS:**

- Poorly matrix conditioning  $\Rightarrow$  **High** number of **solver** iterations.
- No explicit projections  $\Rightarrow$  **Low** number of **nonlinear** iterations.

- **OSS:**

- Better matrix conditioning  $\Rightarrow$  **Low** number of **solver** iterations.
- Explicit projection treatment  $\Rightarrow$  **High** number of **nonlinear** iterations.

# RB-VMS Limitations

- **ASGS:**

- Poorly matrix conditioning  $\Rightarrow$  **High** number of **solver** iterations.
- No explicit projections  $\Rightarrow$  **Low** number of **nonlinear** iterations.

- **OSS:**

- Better matrix conditioning  $\Rightarrow$  **Low** number of **solver** iterations.
- Explicit projection treatment  $\Rightarrow$  **High** number of **nonlinear** iterations.

- **Desired:**

- OSS with implicit projections.

## 1. Motivation

## 2. Residual-based VMS

## 3. Mixed FE VMS

Formulation

Block-preconditioning

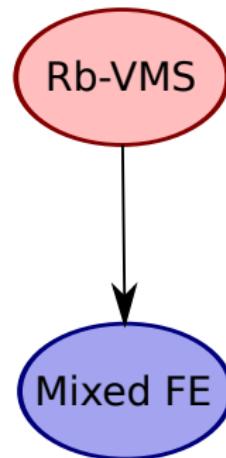
Numerical experiments

Conclusions

## 4. Segregated Runge-Kutta

## 5. Segregated VMS

## 6. Conclusions



# Motivation

## Step by step...

- Residual-based VMS as LES models.
- Mixed FE formulations LES.
- High-order FE methods.
- High-order time integration schemes.
- Adaptive time stepping techniques.
- Velocity-pressure segregation.
- Scalable solvers.
- Application.

# Semidiscrete problem

## FEM equations

$$\begin{aligned}
 & (\mathbf{v}_h, \partial_t \mathbf{u}_h)_\Omega + b(\mathbf{a}, \mathbf{u}_h, \mathbf{v}_h) + (\nabla \mathbf{v}_h, \nu \nabla \mathbf{u}_h)_\Omega - (\nabla \cdot \mathbf{v}_h, p_h)_\Omega \\
 & + (\mathbf{v}_h, \partial_t \tilde{\mathbf{u}})_\Omega + (\mathcal{L}^* \mathbf{v}_h, \tilde{\mathbf{u}})_{\Omega^h} - (\nabla \cdot \mathbf{v}_h, \tilde{p})_{\Omega^h} = \langle \mathbf{v}_h, \mathbf{f} \rangle_\Omega \\
 & (q_h, \nabla \cdot \mathbf{u}_h)_\Omega - (\nabla q_h, \tilde{\mathbf{u}})_{\Omega^h} = 0
 \end{aligned}$$

## SGS equations

$$\partial_t \tilde{\mathbf{u}} + \tau_m^{-1} \tilde{\mathbf{u}} = \mathcal{P} \mathbf{R}_m$$

$$\tau_c^{-1} \tilde{p} = \mathcal{P} R_c$$

$$\mathcal{P} = I \quad (\text{ASGS}), \quad \mathcal{P} = \mathbf{P}_h^\perp = \mathbf{I} - \mathbf{P}_h \quad (\text{OSS})$$

$$\mathbf{a} = \mathbf{u}_h + \tilde{\mathbf{u}}$$

# Semidiscrete problem

## FEM equations

$$\begin{aligned}
 & (\mathbf{v}_h, \partial_t \mathbf{u}_h)_\Omega + b(\mathbf{a}, \mathbf{u}_h, \mathbf{v}_h) + (\nabla \mathbf{v}_h, \nu \nabla \mathbf{u}_h)_\Omega - (\nabla \cdot \mathbf{v}_h, p_h)_\Omega \\
 & + (\mathcal{L}^* \mathbf{v}_h, \tilde{\mathbf{u}})_{\Omega^h} - (\nabla \cdot \mathbf{v}_h, \tilde{p})_{\Omega^h} = \langle \mathbf{v}_h, \mathbf{f} \rangle_\Omega \\
 & (q_h, \nabla \cdot \mathbf{u}_h)_\Omega - (\nabla q_h, \tilde{\mathbf{u}})_{\Omega^h} = 0
 \end{aligned}$$

## SGS equations

$$\tau_m^{-1} \tilde{\mathbf{u}} = \mathcal{P} \mathbf{R}_m$$

$$\tau_c^{-1} \tilde{p} = \mathcal{P} R_c$$

$$\mathcal{P} = \mathbf{P}_h^\perp = \mathbf{I} - \mathbf{P}_h \quad (\text{OSS})$$

$$\mathbf{a} = \mathbf{u}_h$$

# Term-by-term OSS

## FEM equations

$$\begin{aligned}
 & (\mathbf{v}_h, \partial_t \mathbf{u}_h)_\Omega + b(\mathbf{a}, \mathbf{u}_h, \mathbf{v}_h) + (\nabla \mathbf{v}_h, \nu \nabla \mathbf{u}_h)_\Omega - (\nabla \cdot \mathbf{v}_h, p_h)_\Omega \\
 & + (\mathcal{L}^* \mathbf{v}_h, \tilde{\mathbf{u}})_{\Omega^h} - (\nabla \cdot \mathbf{v}_h, \tilde{p})_{\Omega^h} = \langle \mathbf{v}_h, \mathbf{f} \rangle_\Omega \\
 & (q_h, \nabla \cdot \mathbf{u}_h)_\Omega - (\nabla q_h, \tilde{\mathbf{u}})_{\Omega^h} = 0
 \end{aligned}$$

# Term-by-term OSS

## Term-by-term OSS (Codina 2008)

$$(\mathbf{v}_h, \partial_t \mathbf{u}_h)_\Omega + b(\mathbf{a}, \mathbf{u}_h, \mathbf{v}_h) + (\nabla \mathbf{v}_h, \nu \nabla \mathbf{u}_h)_\Omega - (\nabla \cdot \mathbf{v}_h, p_h)_\Omega$$

$$+ \left( \tau_m \mathbf{a} \cdot \nabla \mathbf{v}_h, \mathcal{P}_h^\perp(\mathbf{a} \cdot \nabla \mathbf{u}_h) \right)_{\Omega^h}$$

$$+ (\tau_c \nabla \cdot \mathbf{v}_h, \nabla \cdot \mathbf{u}_h)_{\Omega^h} = \langle \mathbf{v}_h, \mathbf{f} \rangle_\Omega$$

$$(q_h, \nabla \cdot \mathbf{u}_h)_\Omega + \left( \tau_m \nabla q_h, \mathcal{P}_h^\perp(\nabla p_h) \right)_{\Omega^h} = 0$$

# Term-by-term OSS

## Term-by-term OSS (Codina 2008)

$$(\mathbf{v}_h, \partial_t \mathbf{u}_h)_\Omega + b(\mathbf{a}, \mathbf{u}_h, \mathbf{v}_h) + (\nabla \mathbf{v}_h, \nu \nabla \mathbf{u}_h)_\Omega - (\nabla \cdot \mathbf{v}_h, p_h)_\Omega$$

$$+ (\tau_m \mathbf{a} \cdot \nabla \mathbf{v}_h, \mathbf{a} \cdot \nabla \mathbf{u}_h)_{\Omega^h} - (\tau_m \mathbf{a} \cdot \nabla \mathbf{v}_h, \boldsymbol{\eta}_h)_{\Omega^h}$$

$$+ (\tau_c \nabla \cdot \mathbf{v}_h, \nabla \cdot \mathbf{u}_h)_{\Omega^h} = \langle \mathbf{v}_h, \mathbf{f} \rangle_\Omega$$

$$(q_h, \nabla \cdot \mathbf{u}_h)_\Omega + (\tau_m \nabla q_h, \nabla p_h)_{\Omega^h} - (\tau_m \nabla q_h, \boldsymbol{\xi}_h)_{\Omega^h} = 0$$

$$\boldsymbol{\eta}_h := \mathcal{P}_h(\mathbf{a} \cdot \nabla \mathbf{u}_h)$$

$$\boldsymbol{\xi}_h := \mathcal{P}_h(\nabla p_h)$$

# Matrix form

- ASGS:

$$\begin{bmatrix} M\dot{\mathbf{U}} \\ \mathbf{0} \end{bmatrix} + \begin{bmatrix} K + C + A_\tau & G + G_\tau \\ D + D_\tau & L_\tau \end{bmatrix} \begin{bmatrix} \mathbf{U} \\ \mathbf{P} \end{bmatrix} = \begin{bmatrix} \mathbf{F}_u \\ \mathbf{0} \end{bmatrix},$$

# Matrix form

- ASGS:

$$\begin{bmatrix} M\dot{\mathbf{U}} \\ \mathbf{0} \end{bmatrix} + \begin{bmatrix} K + C + A_\tau & G + G_\tau \\ D + D_\tau & L_\tau \end{bmatrix} \begin{bmatrix} \mathbf{U} \\ \mathbf{P} \end{bmatrix} = \begin{bmatrix} \mathbf{F}_u \\ \mathbf{0} \end{bmatrix},$$

- Term-by-term OSS:

$$\begin{bmatrix} M\dot{\mathbf{U}} \\ \mathbf{0} \\ \mathbf{0} \\ \mathbf{0} \end{bmatrix} + \begin{bmatrix} K + C + A_\tau & G & B_{\eta,\tau} & 0 \\ D & L_\tau & 0 & B_{\xi,\tau} \\ -B_{\eta,\tau}^T & 0 & M_{\eta,\tau} & 0 \\ 0 & -B_{\xi,\tau}^T & 0 & M_{\xi,\tau} \end{bmatrix} \begin{bmatrix} \mathbf{U} \\ \mathbf{P} \\ \boldsymbol{\Upsilon} \\ \boldsymbol{\Xi} \end{bmatrix} = \begin{bmatrix} \mathbf{F}_u \\ \mathbf{0} \\ \mathbf{0} \\ \mathbf{0} \end{bmatrix},$$

# Matrix form

- ASGS:

$$\begin{bmatrix} M\dot{\mathbf{U}} \\ \mathbf{0} \end{bmatrix} + \begin{bmatrix} K + C + A_\tau & G + G_\tau \\ D + D_\tau & L_\tau \end{bmatrix} \begin{bmatrix} \mathbf{U} \\ \mathbf{P} \end{bmatrix} = \begin{bmatrix} \mathbf{F}_u \\ \mathbf{0} \end{bmatrix},$$

- Term-by-term OSS:

$$\begin{bmatrix} M\dot{\mathbf{U}} \\ \mathbf{0} \\ \mathbf{0} \\ \mathbf{0} \end{bmatrix} + \begin{bmatrix} K + C + A_\tau & G & B_{\eta,\tau} & 0 \\ D & L_\tau & 0 & B_{\xi,\tau} \\ -B_{\eta,\tau}^T & 0 & M_{\eta,\tau} & 0 \\ 0 & -B_{\xi,\tau}^T & 0 & M_{\xi,\tau} \end{bmatrix} \begin{bmatrix} \mathbf{U} \\ \mathbf{P} \\ \boldsymbol{\Upsilon} \\ \boldsymbol{\Xi} \end{bmatrix} = \begin{bmatrix} \mathbf{F}_u \\ \mathbf{0} \\ \mathbf{0} \\ \mathbf{0} \end{bmatrix},$$

- Term-by-term OSS with Inf-sup stable elements (mixed FE):

$$\begin{bmatrix} M\dot{\mathbf{U}} \\ \mathbf{0} \\ \mathbf{0} \end{bmatrix} + \begin{bmatrix} K + C + A_\tau & G & B_\tau \\ D & 0 & 0 \\ -B_\tau^T & 0 & M_\tau \end{bmatrix} \begin{bmatrix} \mathbf{U} \\ \mathbf{P} \\ \boldsymbol{\Upsilon} \end{bmatrix} = \begin{bmatrix} \mathbf{F}_u \\ \mathbf{0} \\ \mathbf{0} \end{bmatrix},$$

# Matrix form

- ASGS:

$$\begin{bmatrix} M\dot{\mathbf{U}} \\ \mathbf{0} \end{bmatrix} + \begin{bmatrix} K + C + A_\tau & G + G_\tau \\ D + D_\tau & L_\tau \end{bmatrix} \begin{bmatrix} \mathbf{U} \\ \mathbf{P} \end{bmatrix} = \begin{bmatrix} \mathbf{F}_u \\ \mathbf{0} \end{bmatrix},$$

- Term-by-term OSS:

$$\begin{bmatrix} M\dot{\mathbf{U}} \\ \mathbf{0} \\ \mathbf{0} \\ \mathbf{0} \end{bmatrix} + \begin{bmatrix} K + C + A_\tau & G & B_{\eta,\tau} & 0 \\ D & L_\tau & 0 & B_{\xi,\tau} \\ -B_{\eta,\tau}^T & 0 & M_{\eta,\tau} & 0 \\ 0 & -B_{\xi,\tau}^T & 0 & M_{\xi,\tau} \end{bmatrix} \begin{bmatrix} \mathbf{U} \\ \mathbf{P} \\ \boldsymbol{\Upsilon} \\ \boldsymbol{\Xi} \end{bmatrix} = \begin{bmatrix} \mathbf{F}_u \\ \mathbf{0} \\ \mathbf{0} \\ \mathbf{0} \end{bmatrix},$$

- Term-by-term OSS with Inf-sup stable elements (mixed FE):

$$\begin{bmatrix} M\dot{\mathbf{U}} \\ \mathbf{0} \\ \mathbf{0} \end{bmatrix} + \begin{bmatrix} K + C + A_\tau & G & B_\tau \\ D & \mathbf{0} & 0 \\ -B_\tau^T & 0 & M_\tau \end{bmatrix} \begin{bmatrix} \mathbf{U} \\ \mathbf{P} \\ \boldsymbol{\Upsilon} \end{bmatrix} = \begin{bmatrix} \mathbf{F}_u \\ \mathbf{0} \\ \mathbf{0} \end{bmatrix},$$

Index-2 DAE!!!

# Block-recursive preconditioning

General case:

$$\begin{bmatrix} M\dot{\mathbf{U}} \\ \mathbf{0} \\ \mathbf{0} \\ \mathbf{0} \end{bmatrix} + \begin{bmatrix} K + C + A_\tau & G & B_{\eta,\tau} & 0 \\ D & L_\tau & 0 & B_{\xi,\tau} \\ -B_{\eta,\tau}^T & 0 & M_{\eta,\tau} & 0 \\ 0 & -B_{\xi,\tau}^T & 0 & M_{\xi,\tau} \end{bmatrix} \begin{bmatrix} \mathbf{U} \\ \mathbf{P} \\ \boldsymbol{\Upsilon} \\ \boldsymbol{\Xi} \end{bmatrix} = \begin{bmatrix} \mathbf{F}_u \\ \mathbf{0} \\ \mathbf{0} \\ \mathbf{0} \end{bmatrix},$$

# Block-recursive preconditioning

General case:

$$\begin{bmatrix} M\dot{\mathbf{U}} \\ \mathbf{0} \\ \mathbf{0} \\ \mathbf{0} \end{bmatrix} + \begin{bmatrix} K + C + A_\tau & G & | & B_{\eta,\tau} & 0 \\ D & L_\tau & | & 0 & B_{\xi,\tau} \\ -B_{\eta,\tau}^T & 0 & | & M_{\eta,\tau} & 0 \\ 0 & -B_{\xi,\tau}^T & | & 0 & M_{\xi,\tau} \end{bmatrix} \begin{bmatrix} \mathbf{U} \\ \mathbf{P} \\ \boldsymbol{\Upsilon} \\ \boldsymbol{\Xi} \end{bmatrix} = \begin{bmatrix} \mathbf{F}_u \\ \mathbf{0} \\ \mathbf{0} \\ \mathbf{0} \end{bmatrix},$$

# Block-recursive preconditioning

General case:

$$\begin{bmatrix} M\dot{\mathbf{U}} \\ \mathbf{0} \\ \mathbf{0} \\ \mathbf{0} \end{bmatrix} + \begin{bmatrix} K + C + A_\tau & G & | & B_{\eta,\tau} & 0 \\ D & L_\tau & | & 0 & B_{\xi,\tau} \\ -B_{\eta,\tau}^T & 0 & | & M_{\eta,\tau} & 0 \\ 0 & -B_{\xi,\tau}^T & | & 0 & M_{\xi,\tau} \end{bmatrix} \begin{bmatrix} \mathbf{U} \\ \mathbf{P} \\ \boldsymbol{\Upsilon} \\ \Xi \end{bmatrix} = \begin{bmatrix} \mathbf{F}_u \\ \mathbf{0} \\ \mathbf{0} \\ \mathbf{0} \end{bmatrix},$$

- Outer block matrix:

$$\tilde{A} = \begin{bmatrix} M_{\eta,\tau} & 0 & | & -B_{\eta,\tau}^T & 0 \\ 0 & M_{\xi,\tau} & | & 0 & -B_{\xi,\tau}^T \\ B_{\eta,\tau} & 0 & | & K + C + A_\tau & G \\ 0 & B_{\xi,\tau} & | & D & L_\tau \end{bmatrix}$$

# Block-recursive preconditioning

General case:

$$\begin{bmatrix} M\dot{\mathbf{U}} \\ \mathbf{0} \\ \mathbf{0} \\ \mathbf{0} \end{bmatrix} + \begin{bmatrix} K + C + A_\tau & G & | & B_{\eta,\tau} & 0 \\ D & L_\tau & | & 0 & B_{\xi,\tau} \\ -B_{\eta,\tau}^T & 0 & | & M_{\eta,\tau} & 0 \\ 0 & -B_{\xi,\tau}^T & | & 0 & M_{\xi,\tau} \end{bmatrix} \begin{bmatrix} \mathbf{U} \\ \mathbf{P} \\ \boldsymbol{\tau} \\ \boldsymbol{\Xi} \end{bmatrix} = \begin{bmatrix} \mathbf{F}_u \\ \mathbf{0} \\ \mathbf{0} \\ \mathbf{0} \end{bmatrix},$$

- Outer block matrix:

# Block-recursive preconditioning

General case:

$$\begin{bmatrix} M\dot{\mathbf{U}} \\ \mathbf{0} \\ \mathbf{0} \\ \mathbf{0} \end{bmatrix} + \begin{bmatrix} K + C + A_\tau & G & | & B_{\eta,\tau} & 0 \\ D & L_\tau & | & 0 & B_{\xi,\tau} \\ -B_{\eta,\tau}^T & 0 & | & M_{\eta,\tau} & 0 \\ 0 & -B_{\xi,\tau}^T & | & 0 & M_{\xi,\tau} \end{bmatrix} \begin{bmatrix} \mathbf{U} \\ \mathbf{P} \\ \boldsymbol{\Upsilon} \\ \boldsymbol{\Xi} \end{bmatrix} = \begin{bmatrix} \mathbf{F}_u \\ \mathbf{0} \\ \mathbf{0} \\ \mathbf{0} \end{bmatrix},$$

- Outer block matrix:

$$\tilde{A} = \begin{bmatrix} \tilde{M}_\tau & -\tilde{B}_\tau^T \\ \tilde{B}_\tau & \tilde{K}_\tau \end{bmatrix} = \begin{bmatrix} I & 0 \\ \tilde{B}_\tau \tilde{M}_\tau^{-1} & I \end{bmatrix} \begin{bmatrix} \tilde{M}_\tau & -\tilde{B}_\tau^T \\ 0 & \tilde{S} \end{bmatrix},$$

# Block-recursive preconditioning

General case:

$$\begin{bmatrix} M\dot{\mathbf{U}} \\ \mathbf{0} \\ \mathbf{0} \\ \mathbf{0} \end{bmatrix} + \begin{bmatrix} K + C + A_\tau & G & | & B_{\eta,\tau} & 0 \\ D & L_\tau & | & 0 & B_{\xi,\tau} \\ -B_{\eta,\tau}^T & 0 & | & M_{\eta,\tau} & 0 \\ 0 & -B_{\xi,\tau}^T & | & 0 & M_{\xi,\tau} \end{bmatrix} \begin{bmatrix} \mathbf{U} \\ \mathbf{P} \\ \boldsymbol{\Upsilon} \\ \boldsymbol{\Xi} \end{bmatrix} = \begin{bmatrix} \mathbf{F}_u \\ \mathbf{0} \\ \mathbf{0} \\ \mathbf{0} \end{bmatrix},$$

- Outer block matrix:

$$\begin{aligned} \tilde{A} &= \begin{bmatrix} \tilde{M}_\tau & -\tilde{B}_\tau^T \\ \tilde{B}_\tau & \tilde{K}_\tau \end{bmatrix} = \begin{bmatrix} I & 0 \\ \tilde{B}_\tau \tilde{M}_\tau^{-1} & I \end{bmatrix} \begin{bmatrix} \tilde{M}_\tau & -\tilde{B}_\tau^T \\ 0 & \tilde{S} \end{bmatrix}, \\ \tilde{S} &= \tilde{K}_\tau + \tilde{B}_\tau \tilde{M}_\tau^{-1} \tilde{B}_\tau^T, \end{aligned}$$

# Block-recursive preconditioning

General case:

$$\begin{bmatrix} M\dot{\mathbf{U}} \\ \mathbf{0} \\ \mathbf{0} \\ \mathbf{0} \end{bmatrix} + \begin{bmatrix} K + C + A_\tau & G & | & B_{\eta,\tau} & 0 \\ D & L_\tau & | & 0 & B_{\xi,\tau} \\ -B_{\eta,\tau}^T & 0 & | & M_{\eta,\tau} & 0 \\ 0 & -B_{\xi,\tau}^T & | & 0 & M_{\xi,\tau} \end{bmatrix} \begin{bmatrix} \mathbf{U} \\ \mathbf{P} \\ \boldsymbol{\Upsilon} \\ \Xi \end{bmatrix} = \begin{bmatrix} \mathbf{F}_u \\ \mathbf{0} \\ \mathbf{0} \\ \mathbf{0} \end{bmatrix},$$

- Outer block matrix:

$$\tilde{A} = \begin{bmatrix} \tilde{M}_\tau & -\tilde{B}_\tau^T \\ \tilde{B}_\tau & \tilde{K}_\tau \end{bmatrix} = \begin{bmatrix} I & 0 \\ \tilde{B}_\tau \tilde{M}_\tau^{-1} & I \end{bmatrix} \begin{bmatrix} \tilde{M}_\tau & -\tilde{B}_\tau^T \\ 0 & \tilde{S} \end{bmatrix},$$

$$\tilde{S} = \tilde{K}_\tau + \tilde{B}_\tau \tilde{M}_\tau^{-1} \tilde{B}_\tau^T,$$

$$P_U(\tilde{A}) = \begin{bmatrix} \tilde{M}_\tau & -\tilde{B}_\tau^T \\ 0 & \tilde{S} \end{bmatrix}^{-1} = \begin{bmatrix} \tilde{M}_\tau^{-1} & \tilde{M}_\tau^{-1} \tilde{B}_\tau^T \tilde{S}^{-1} \\ 0 & \tilde{S}^{-1} \end{bmatrix},$$

# Block-recursive preconditioning

General case:

$$\begin{bmatrix} M\dot{\mathbf{U}} \\ \mathbf{0} \\ \mathbf{0} \\ \mathbf{0} \end{bmatrix} + \begin{bmatrix} K + C + A_\tau & G & B_{\eta,\tau} & 0 \\ D & L_\tau & 0 & B_{\xi,\tau} \\ -B_{\eta,\tau}^T & 0 & M_{\eta,\tau} & 0 \\ 0 & -B_{\xi,\tau}^T & 0 & M_{\xi,\tau} \end{bmatrix} \begin{bmatrix} \mathbf{U} \\ \mathbf{P} \\ \boldsymbol{\Upsilon} \\ \boldsymbol{\Xi} \end{bmatrix} = \begin{bmatrix} \mathbf{F}_u \\ \mathbf{0} \\ \mathbf{0} \\ \mathbf{0} \end{bmatrix},$$

- Outer block matrix:

$$\tilde{A} = \begin{bmatrix} \tilde{M}_\tau & -\tilde{B}_\tau^T \\ \tilde{B}_\tau & \tilde{K}_\tau \end{bmatrix} = \begin{bmatrix} I & 0 \\ \tilde{B}_\tau \tilde{M}_\tau^{-1} & I \end{bmatrix} \begin{bmatrix} \tilde{M}_\tau & -\tilde{B}_\tau^T \\ 0 & \tilde{S} \end{bmatrix},$$

$$\tilde{S} = \tilde{K}_\tau + \tilde{B}_\tau \tilde{M}_\tau^{-1} \tilde{B}_\tau^T,$$

$$P_U(\tilde{A}) = \begin{bmatrix} \tilde{M}_\tau & -\tilde{B}_\tau^T \\ 0 & \tilde{S} \end{bmatrix}^{-1} = \begin{bmatrix} \tilde{M}_\tau^{-1} & \tilde{M}_\tau^{-1} \tilde{B}_\tau^T \tilde{S}^{-1} \\ 0 & \tilde{S}^{-1} \end{bmatrix},$$

$$\tilde{S}^{-1} \approx \tilde{K}_\tau^{-1}.$$

# Block-recursive preconditioning

General case:

$$\begin{bmatrix} M\dot{\mathbf{U}} \\ \mathbf{0} \\ \mathbf{0} \\ \mathbf{0} \end{bmatrix} + \begin{bmatrix} K + C + A_\tau & G & B_{\eta,\tau} & 0 \\ D & L_\tau & 0 & B_{\xi,\tau} \\ -B_{\eta,\tau}^T & 0 & M_{\eta,\tau} & 0 \\ 0 & -B_{\xi,\tau}^T & 0 & M_{\xi,\tau} \end{bmatrix} \begin{bmatrix} \mathbf{U} \\ \mathbf{P} \\ \boldsymbol{\Upsilon} \\ \boldsymbol{\Xi} \end{bmatrix} = \begin{bmatrix} \mathbf{F}_u \\ \mathbf{0} \\ \mathbf{0} \\ \mathbf{0} \end{bmatrix},$$

- Inner block matrix:

$$\tilde{K}_\tau = \begin{bmatrix} K_\tau & G \\ D & L_\tau \end{bmatrix} = \begin{bmatrix} I & 0 \\ DK_\tau^{-1} & I \end{bmatrix} \begin{bmatrix} K_\tau & G \\ 0 & S \end{bmatrix},$$

# Block-recursive preconditioning

General case:

$$\begin{bmatrix} M\dot{\mathbf{U}} \\ \mathbf{0} \\ \mathbf{0} \\ \mathbf{0} \end{bmatrix} + \begin{bmatrix} K + C + A_\tau & G & B_{\eta,\tau} & 0 \\ D & L_\tau & 0 & B_{\xi,\tau} \\ -B_{\eta,\tau}^T & 0 & M_{\eta,\tau} & 0 \\ 0 & -B_{\xi,\tau}^T & 0 & M_{\xi,\tau} \end{bmatrix} \begin{bmatrix} \mathbf{U} \\ \mathbf{P} \\ \boldsymbol{\Upsilon} \\ \boldsymbol{\Xi} \end{bmatrix} = \begin{bmatrix} \mathbf{F}_u \\ \mathbf{0} \\ \mathbf{0} \\ \mathbf{0} \end{bmatrix},$$

- Inner block matrix:

$$\begin{aligned} \tilde{K}_\tau &= \begin{bmatrix} K_\tau & G \\ D & L_\tau \end{bmatrix} = \begin{bmatrix} I & 0 \\ DK_\tau^{-1} & I \end{bmatrix} \begin{bmatrix} K_\tau & G \\ 0 & S \end{bmatrix}, \\ S &= L_\tau - DK_\tau^{-1}G, \end{aligned}$$

# Block-recursive preconditioning

General case:

$$\begin{bmatrix} M\dot{\mathbf{U}} \\ \mathbf{0} \\ \mathbf{0} \\ \mathbf{0} \end{bmatrix} + \begin{bmatrix} K + C + A_\tau & G & B_{\eta,\tau} & 0 \\ D & L_\tau & 0 & B_{\xi,\tau} \\ -B_{\eta,\tau}^T & 0 & M_{\eta,\tau} & 0 \\ 0 & -B_{\xi,\tau}^T & 0 & M_{\xi,\tau} \end{bmatrix} \begin{bmatrix} \mathbf{U} \\ \mathbf{P} \\ \boldsymbol{\Upsilon} \\ \boldsymbol{\Xi} \end{bmatrix} = \begin{bmatrix} \mathbf{F}_u \\ \mathbf{0} \\ \mathbf{0} \\ \mathbf{0} \end{bmatrix},$$

- Inner block matrix:

$$\tilde{K}_\tau = \begin{bmatrix} K_\tau & G \\ D & L_\tau \end{bmatrix} = \begin{bmatrix} I & 0 \\ DK_\tau^{-1} & I \end{bmatrix} \begin{bmatrix} K_\tau & G \\ 0 & S \end{bmatrix},$$

$$S = L_\tau - DK_\tau^{-1}G,$$

$$P_U(\tilde{K}_\tau) = \begin{bmatrix} K_\tau & G \\ 0 & S \end{bmatrix}^{-1} = \begin{bmatrix} K_\tau^{-1} & -K_\tau^{-1}GS^{-1} \\ 0 & S^{-1} \end{bmatrix},$$

# Block-recursive preconditioning

General case:

$$\begin{bmatrix} M\dot{\mathbf{U}} \\ \mathbf{0} \\ \mathbf{0} \\ \mathbf{0} \end{bmatrix} + \begin{bmatrix} K + C + A_\tau & G & B_{\eta,\tau} & 0 \\ D & L_\tau & 0 & B_{\xi,\tau} \\ -B_{\eta,\tau}^T & 0 & M_{\eta,\tau} & 0 \\ 0 & -B_{\xi,\tau}^T & 0 & M_{\xi,\tau} \end{bmatrix} \begin{bmatrix} \mathbf{U} \\ \mathbf{P} \\ \boldsymbol{\Upsilon} \\ \boldsymbol{\Xi} \end{bmatrix} = \begin{bmatrix} \mathbf{F}_u \\ \mathbf{0} \\ \mathbf{0} \\ \mathbf{0} \end{bmatrix},$$

- Inner block matrix:

$$\tilde{K}_\tau = \begin{bmatrix} K_\tau & G \\ D & L_\tau \end{bmatrix} = \begin{bmatrix} I & 0 \\ DK_\tau^{-1} & I \end{bmatrix} \begin{bmatrix} K_\tau & G \\ 0 & S \end{bmatrix},$$

$$S = L_\tau - DK_\tau^{-1}G,$$

$$P_U(\tilde{K}_\tau) = \begin{bmatrix} K_\tau & G \\ 0 & S \end{bmatrix}^{-1} = \begin{bmatrix} K_\tau^{-1} & -K_\tau^{-1}GS^{-1} \\ 0 & S^{-1} \end{bmatrix},$$

$$\tilde{S}^{-1} \approx L_p^{-1}.$$

# Numerical experiments

Manufactured analytical solution:

- Colliding flow.

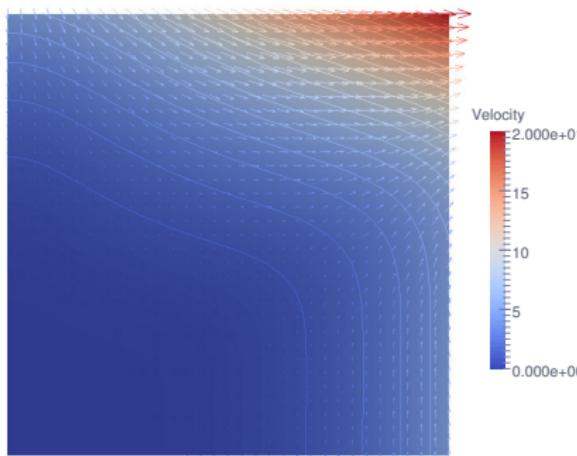
Two different turbulent benchmarks:

- Taylor-Green Vortex (TGV) flow.
- Turbulent Channel Flow (TCF).

# Colliding flow

## Problem setting:

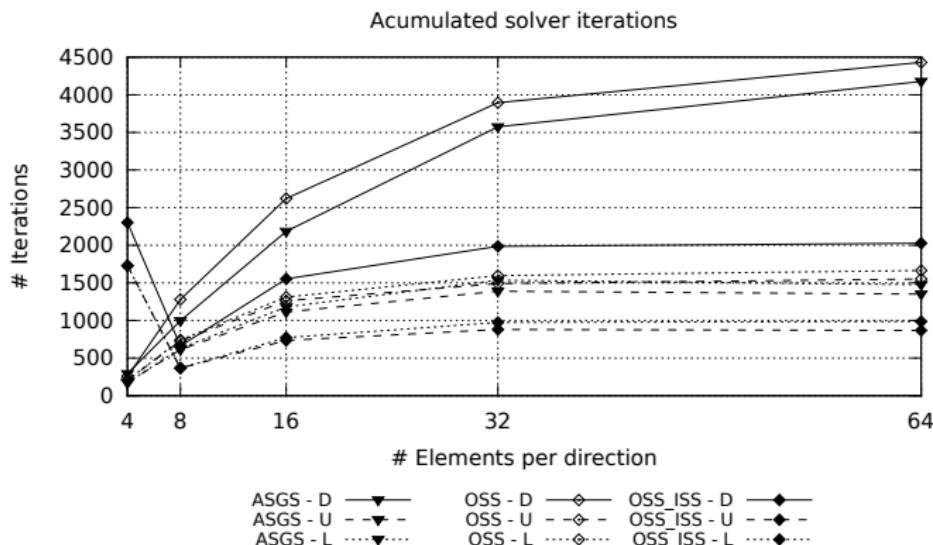
- Analytical solution.
- $Re = 25$ .
- Mesh refinement:  $4^3$  to  $64^3$  Q1/Q1 elements (ASGS and OSS) or  $2^3$  to  $32^3$  Q2/Q1 elements (OSS-ISS)



Velocity field

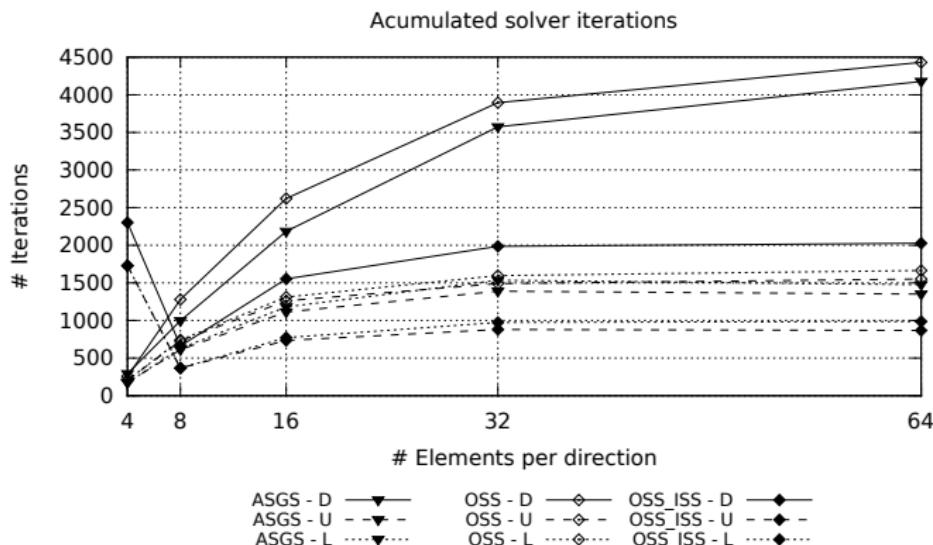
# Colliding flow

**Acumulated solver iterations:** (using  $P_U(\tilde{A})$ )



# Colliding flow

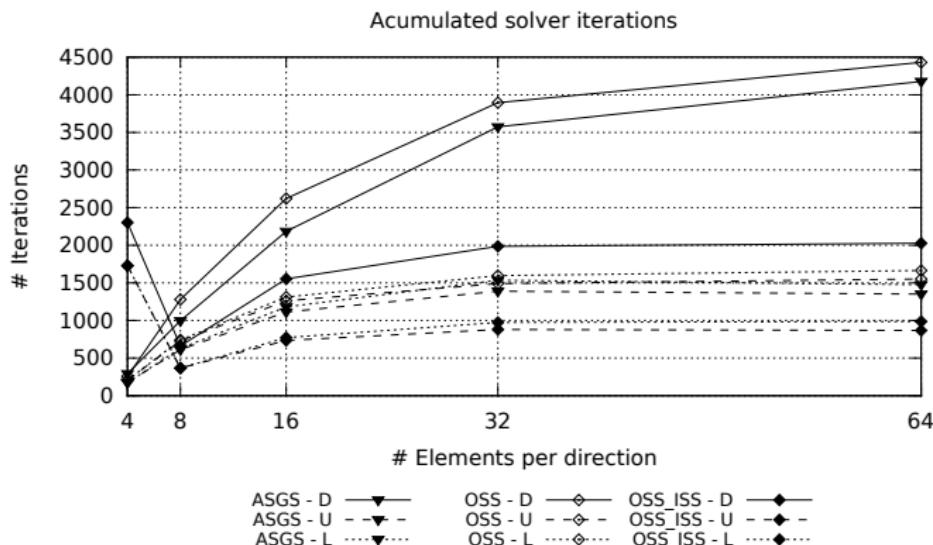
**Acumulated solver iterations:** (using  $P_U(\tilde{A})$ )



- $P_U(\tilde{K}_\tau)$  and  $P_L(\tilde{K}_\tau)$  optimal block-preconditioners for all methods.

# Colliding flow

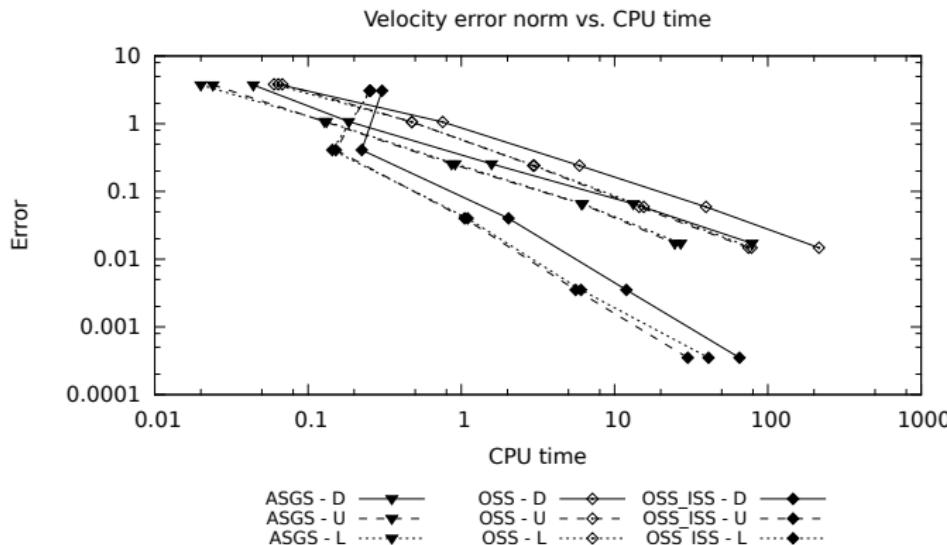
**Acumulated solver iterations:** (using  $P_U(\tilde{A})$ )



- $P_U(\tilde{K}_\tau)$  and  $P_L(\tilde{K}_\tau)$  optimal block-preconditioners for all methods.
- Less solver iterations for the OSS-ISS method with the same velocity DOFs.

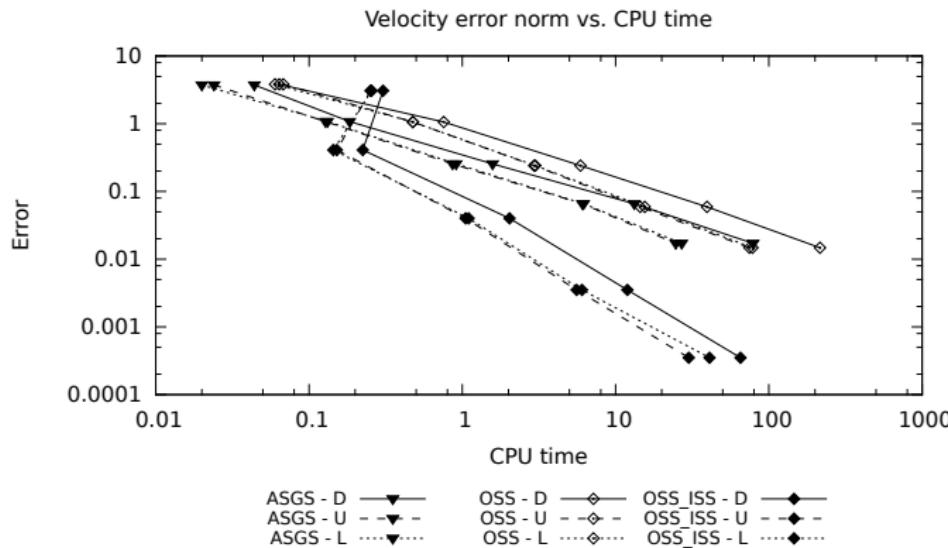
# Colliding flow

## Efficiency: Velocity



# Colliding flow

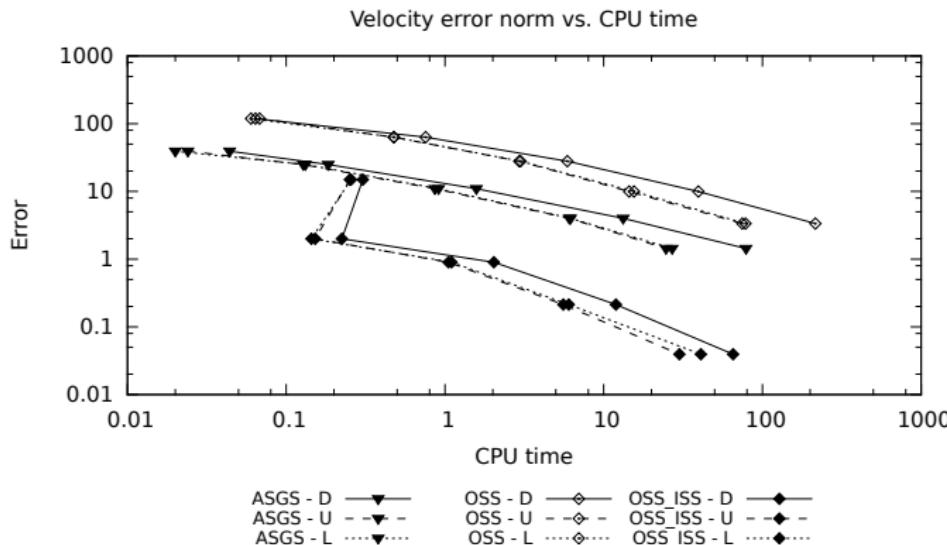
## Efficiency: Velocity



- OSS-ISS the most efficient method.

# Colliding flow

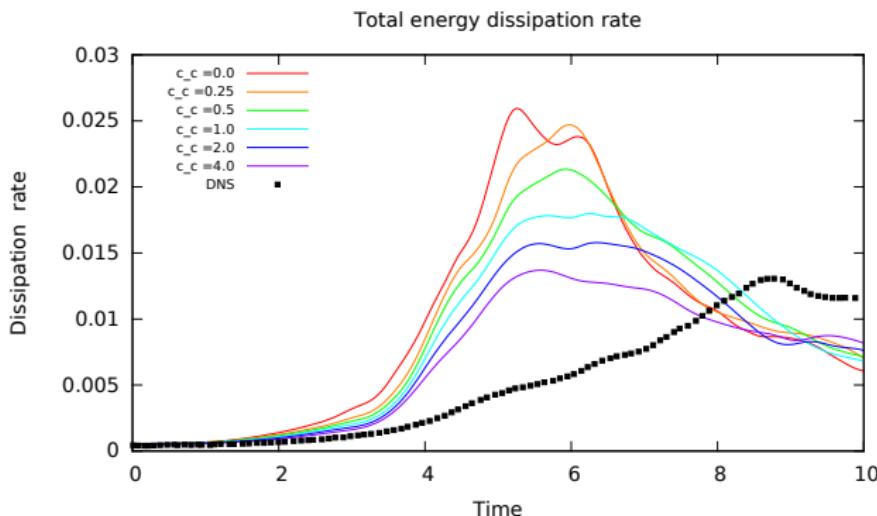
## Efficiency: Pressure



- Also for pressures.

# TGV Taylor-Green Vortex flow

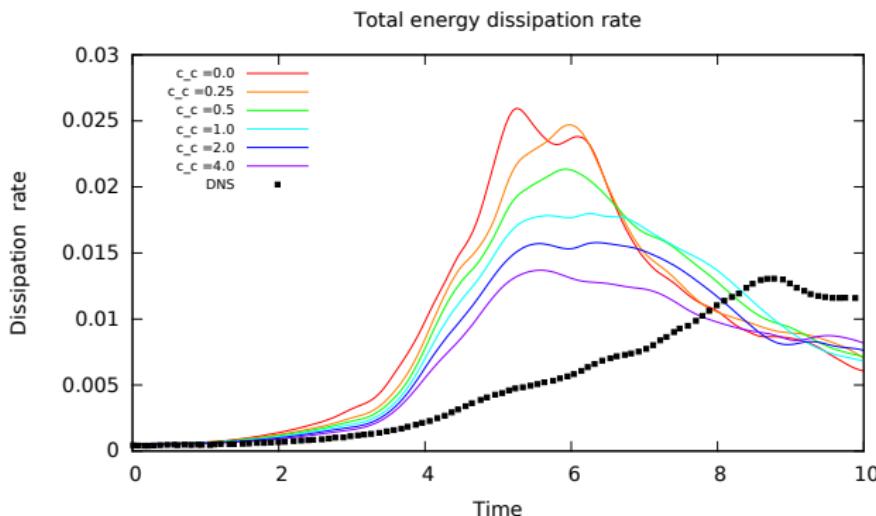
## Effect of the grad-div term ( $\tau_c \nabla \cdot \mathbf{u}, \nabla \cdot \mathbf{v}$ ): (coarse mesh)



Total energy dissipation rate.

# TGV Taylor-Green Vortex flow

## Effect of the grad-div term ( $\tau_c \nabla \cdot \mathbf{u}, \nabla \cdot \mathbf{v}$ ): (coarse mesh)

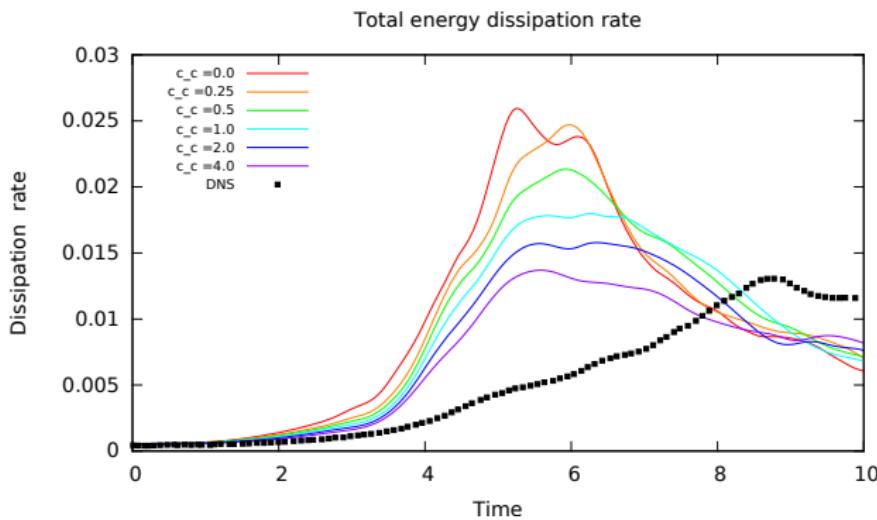


Total energy dissipation rate.

- Bad results when  $c_c \rightarrow 0$ .

# TGV Taylor-Green Vortex flow

## Effect of the grad-div term ( $\tau_c \nabla \cdot \mathbf{u}, \nabla \cdot \mathbf{v}$ ): (coarse mesh)

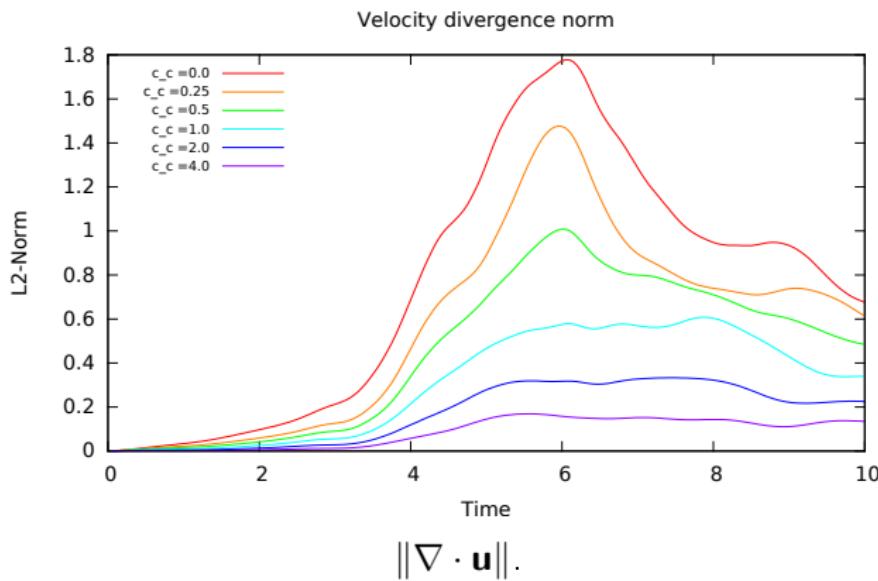


Total energy dissipation rate.

- Bad results when  $c_c \rightarrow 0$ .
- Best option  $c_c = 4.0$ .

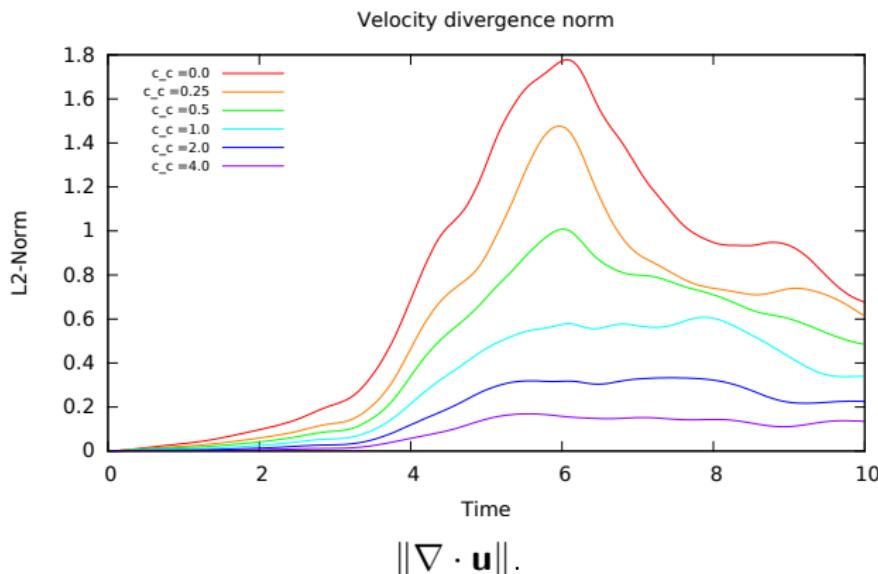
# TGV Taylor-Green Vortex flow

**Effect of the grad-div term ( $\tau_c \nabla \cdot \mathbf{u}, \nabla \cdot \mathbf{v}$ ): (coarse mesh)**



# TGV Taylor-Green Vortex flow

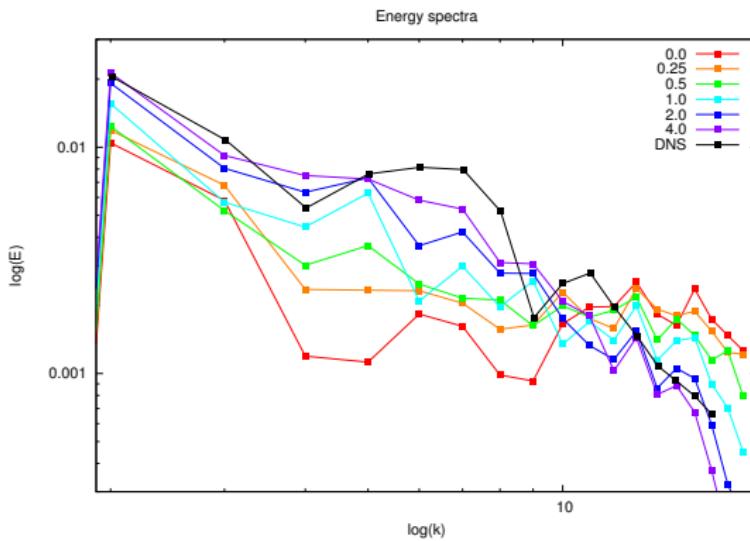
## Effect of the grad-div term ( $\tau_c \nabla \cdot \mathbf{u}, \nabla \cdot \mathbf{v}$ ): (coarse mesh)



- Incompressibility constraint not satisfied.

# TGV Taylor-Green Vortex flow

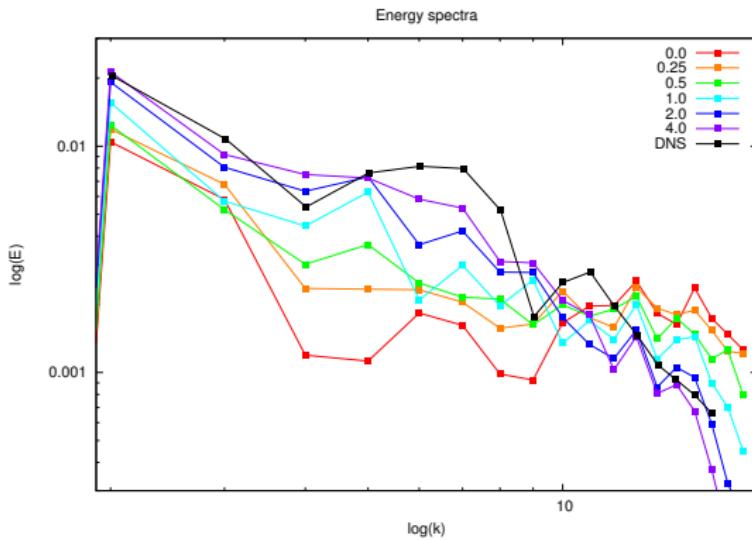
## Effect of the grad-div term ( $\tau_c \nabla \cdot \mathbf{u}, \nabla \cdot \mathbf{v}$ ): (coarse mesh)



Energy spectra.

# TGV Taylor-Green Vortex flow

## Effect of the grad-div term ( $\tau_c \nabla \cdot \mathbf{u}, \nabla \cdot \mathbf{v}$ ): (coarse mesh)

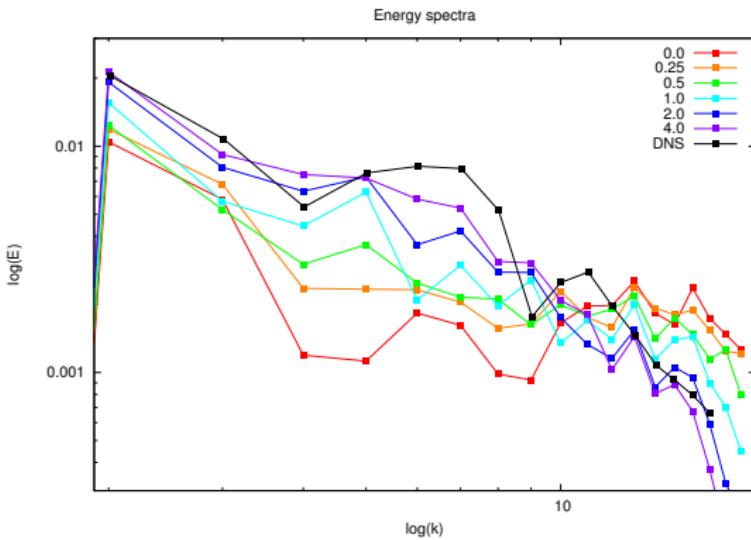


Energy spectra.

- Overdissipation on the large scales when  $c_c \rightarrow 0$ .

# TGV Taylor-Green Vortex flow

## Effect of the grad-div term ( $\tau_c \nabla \cdot \mathbf{u}, \nabla \cdot \mathbf{v}$ ): (coarse mesh)

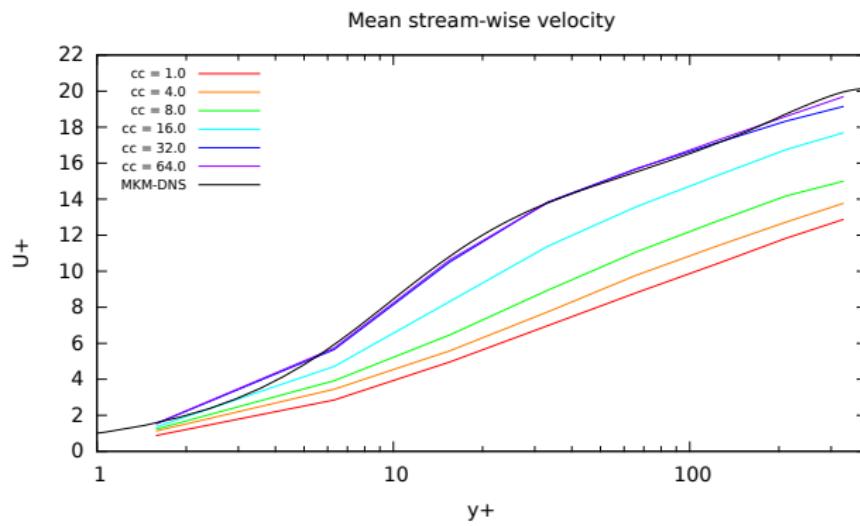


Energy spectra.

- Overdissipation on the large scales when  $c_c \rightarrow 0$ .
- Infradissipation on the small scales when  $c_c \rightarrow 0$ .

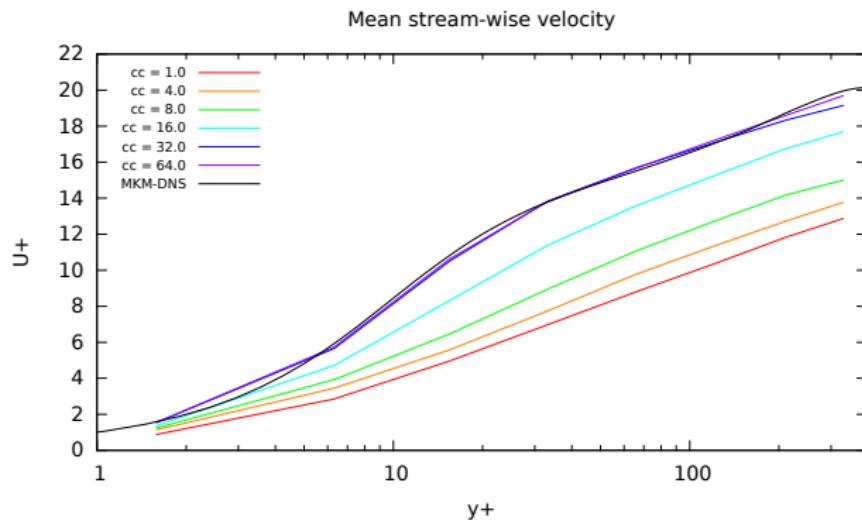
# TCF Turbulent Channel Flow at $Re_\tau = 395$

## Mean streamwise velocity (models):



# TCF Turbulent Channel Flow at $Re_\tau = 395$

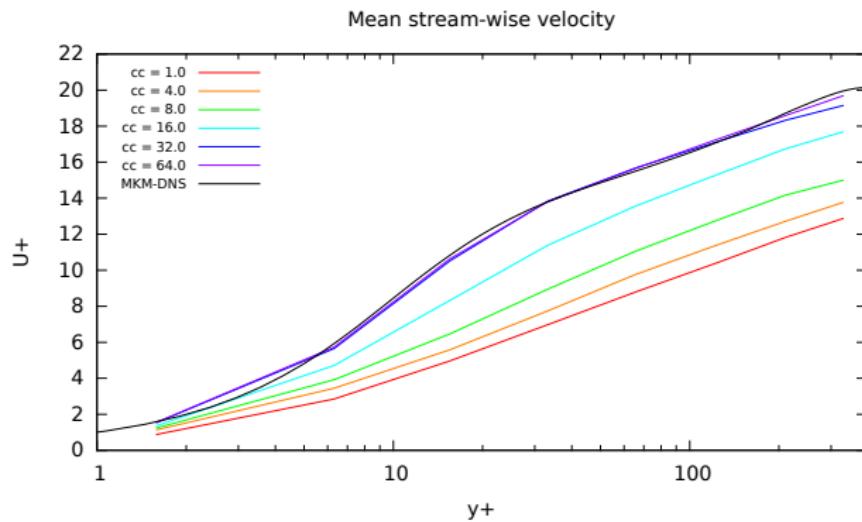
## Mean streamwise velocity (models):



- Same behaviour observed for TCF test.

# TCF Turbulent Channel Flow at $Re_\tau = 395$

## Mean streamwise velocity (models):



- Same behaviour observed for TCF test.
- Best option  $c_c = 32.0$ .

# Mixed FE VMS Conclusions

- Scalable recursive block-preconditioners

# Mixed FE VMS Conclusions

- Scalable recursive block-preconditioners
- Slightly better results for equal-order interpolations

# Mixed FE VMS Conclusions

- Scalable recursive block-preconditioners
- Slightly better results for equal-order interpolations
- Mixed FE OSS the most efficient method

# Mixed FE VMS Conclusions

- Scalable recursive block-preconditioners
- Slightly better results for equal-order interpolations
- Mixed FE OSS the most efficient method
- Mixed FE OSS keeps the index-2 DAE nature in time of the problem

# Mixed FE VMS Limitations

- Strong dependency on the grad-div stabilization term

# Mixed FE VMS Limitations

- Strong dependency on the grad-div stabilization term
- Optimal choice of  $c_c$  is problem dependent

## 1. Motivation

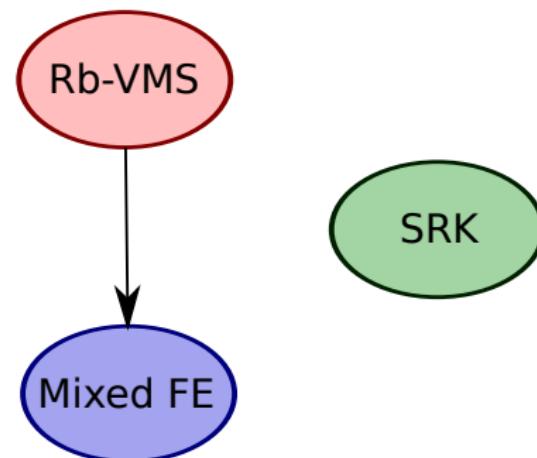
## 2. Residual-based VMS

## 3. Mixed FE VMS

## 4. Segregated Runge-Kutta Formulation Numerical experiments Conclusions

## 5. Segregated VMS

## 6. Conclusions



# Motivation

## Step by step...

- Residual-based VMS as LES models.
- Mixed FE formulations LES.
- High-order FE methods.
- High-order time integration schemes.
- Adaptive time stepping techniques.
- Velocity-pressure segregation.
- Scalable solvers.
- Application.

# Incomp. Navier Stokes equations

Find  $\mathbf{u}$  and  $p$  defined in  $\Omega$

$$\partial_t \mathbf{u} + \mathbf{u} \cdot \nabla \mathbf{u} + \nabla p - \nu \nabla^2 \mathbf{u} = \mathbf{f}$$

$$\nabla \cdot \mathbf{u} = 0$$

with appropriate boundary conditions on  $\Gamma$ .

# Incomp. Navier Stokes equations

Find  $\mathbf{u}$  and  $p$  defined in  $\Omega$

$$\partial_t \mathbf{u} + \mathbf{u} \cdot \nabla \mathbf{u} + \nabla p - \nu \nabla^2 \mathbf{u} = \mathbf{f}$$

$$\nabla \cdot \mathbf{u} = 0$$

with appropriate boundary conditions on  $\Gamma$ .

- **Laminar** case (no convection stabilization required).

# Incomp. Navier Stokes equations

Find  $\mathbf{u}$  and  $p$  defined in  $\Omega$

$$\partial_t \mathbf{u} + \mathbf{u} \cdot \nabla \mathbf{u} + \nabla p - \nu \nabla^2 \mathbf{u} = \mathbf{f}$$

$$\nabla \cdot \mathbf{u} = 0$$

with appropriate boundary conditions on  $\Gamma$ .

- Laminar case (no convection stabilization required).
- Saddle-point structure (stability via **mixed FE**).

# Incomp. Navier Stokes equations

Find  $\mathbf{u}$  and  $p$  defined in  $\Omega$

$$\begin{aligned}\partial_t \mathbf{u} + \mathbf{u} \cdot \nabla \mathbf{u} + \nabla p - \nu \nabla^2 \mathbf{u} &= \mathbf{f} \\ \nabla \cdot \mathbf{u} &= 0\end{aligned}$$

with appropriate boundary conditions on  $\Gamma$ .

- Laminar case (no convection stabilization required).
- Saddle-point structure (stability via mixed FE).
- Index-2 DAE system (time integration with care)

# Incomp. Navier Stokes equations

Find  $\mathbf{U}$  and  $\mathbf{P}$

$$\begin{aligned} M\dot{\mathbf{U}} + (K + C(\mathbf{U}))\mathbf{U} + G\mathbf{P} &= \mathbf{F}, \\ D\mathbf{U} &= \mathbf{G} \end{aligned}$$

with appropriate boundary conditions on  $\Gamma$ .

- Laminar case (no convection stabilization required).
- Saddle-point structure (stability via mixed FE).
- Index-2 DAE system (time integration with care)

# Velocity-pressure splitting

**Existing methods: Fractional step / pressure segregation methods**

$$\begin{aligned} \dot{\mathbf{U}} + M^{-1}(K + C(\mathbf{U}))\mathbf{U} + M^{-1}G\mathbf{P}^* &= \mathbf{F}, & \text{in } (t^n, t^{n+1}], \\ -DM^{-1}G\mathbf{P} &= DM^{-1}(K + C(\mathbf{U})\mathbf{U} - \mathbf{F}) + \dot{\mathbb{G}}, & \text{in } t^{n+1}, \end{aligned}$$

Some properties:

- $P^*$  extrapolation from previous time steps
- Time-marching scheme for the momentum eq'on (ODE system!)
- Velocity and pressure computations are segregated
- A must for computational efficiency (coercive blocks)
- 2nd order splitting error ( $> 2$  unstable)

# Velocity-pressure splitting

**Existing methods: Fractional step / pressure segregation methods**

$$\begin{aligned} \dot{\mathbf{U}} + M^{-1}(K + C(\mathbf{U}))\mathbf{U} + M^{-1}G\mathbf{P}^* &= \mathbf{F}, & \text{in } (t^n, t^{n+1}], \\ -DM^{-1}G\mathbf{P} &= DM^{-1}(K + C(\mathbf{U})\mathbf{U} - \mathbf{F}) + \dot{\mathbb{G}}, & \text{in } t^{n+1}, \end{aligned}$$

*High-order time integration:*

- Fractional step techniques + (e.g.) Runge-Kutta for momentum equation: [Nikitin'06; Knikker'09; Kampanis et al'06...]
- Introduce segregation error (2nd order)

# Velocity-pressure splitting

**Existing methods: Fractional step / pressure segregation methods**

$$\begin{aligned} \dot{\mathbf{U}} + M^{-1}(K + C(\mathbf{U}))\mathbf{U} + M^{-1}G\mathbf{P}^* &= \mathbf{F}, & \text{in } (t^n, t^{n+1}], \\ -DM^{-1}G\mathbf{P} &= DM^{-1}(K + C(\mathbf{U})\mathbf{U} - \mathbf{F}) + \dot{\mathbb{G}}, & \text{in } t^{n+1}, \end{aligned}$$

*High-order time integration:*

- Fractional step techniques + (e.g.) Runge-Kutta for momentum equation: [Nikitin'06; Knikker'09; Kampanis et al'06...]
- Introduce segregation error (2nd order)

Drawbacks:

- Segregation error can jeopardize high order time integrators
- Effect of segregation on pressure error unclear (not reported in literature)

# Velocity-pressure splitting

Existing methods: Half-explicit RK methods (HERK)

$$\frac{1}{\delta t} \mathbf{U}_i = \frac{1}{\delta t} \mathbf{U}_n + \sum_{j=1}^{i-1} \hat{a}_{ij} M^{-1} (K \mathbf{U}_j + C(\mathbf{U}_j) \mathbf{U}_j + G \mathbf{P}_j),$$

$$D \mathbf{U}_i = \mathbb{G}(t_i),$$

# Velocity-pressure splitting

**Existing methods: Half-explicit RK methods (HERK)**

$$\frac{1}{\delta t} \mathbf{U}_i = \frac{1}{\delta t} \mathbf{U}_n - \sum_{j=1}^{i-1} \hat{a}_{ij} M^{-1} (KU_j + C(U_j)U_j + GP_j),$$

$$D\mathbf{U}_n - \delta t \sum_{j=1}^{i-1} \hat{a}_{ij} DM^{-1} (KU_j + C(U_j)U_j + GP_j) = \mathbb{G}(t_i).$$

*High-order time integration:*

- Recently applied to the Navier-Stokes equations [Sanderse & Koren '12]
- Velocity-pressure segregation by the time integrator
- No additional splitting needed (high-order feasible, not spoiled)

# Velocity-pressure splitting

## Existing methods: Half-explicit RK methods (HERK)

$$\frac{1}{\delta t} \mathbf{U}_i = \frac{1}{\delta t} \mathbf{U}_n - \sum_{j=1}^{i-1} \hat{a}_{ij} M^{-1} (KU_j + C(U_j)U_j + GP_j),$$

$$D\mathbf{U}_n - \delta t \sum_{j=1}^{i-1} \hat{a}_{ij} DM^{-1} (KU_j + C(U_j)U_j + GP_j) = \mathbb{G}(t_i).$$

Drawbacks:

- All terms must be treated explicitly (viscous + convective terms)
- Implicit treatment of interest when  $\delta t_{CFL}$  smaller than finest time scales of interest [Verstappen & Veldman'03; Vreman & Kuerten'14]
- A straightforward application with implicit/IMEX integrators leads to, e.g,  $D(M + \delta t K_V)^{-1} G$  matrix for the pressure (not affordable)

# Segregated RK methods

- **Observation:** No implicit / IMEX high-order integrators with implicit velocity-pressure segregation

# Segregated RK methods

- **Observation:** No implicit / IMEX high-order integrators with implicit velocity-pressure segregation
- **Target:** Develop such algorithms (of RK type)

# Segregated RK methods

- **Observation:** No implicit / IMEX high-order integrators with implicit velocity-pressure segregation
- **Target:** Develop such algorithms (of RK type)
- **Result:** Segregated RK methods [OC & Badia, 2015]

# Segregated RK methods

The idea:

1. Consider the projected momentum eq'on on the discretely divergence free space

$$\begin{aligned} M\dot{\mathbf{U}} + (K + C(\mathbf{U}))\mathbf{U} + G\mathbf{P} &= \mathbf{F}, \\ D\mathbf{U} &= \mathbf{G} \end{aligned}$$

# Segregated RK methods

The idea:

1. Consider the projected momentum eq'on on the discretely divergence free space

$$\begin{aligned} M\dot{\mathbf{U}} + (K + C(\mathbf{U}))\mathbf{U} + G\mathbf{P} &= \mathbf{F}, \\ -DM^{-1}G\mathbf{P} &= DM^{-1}(K + C(\mathbf{U})\mathbf{U} - \mathbf{F}) + \dot{\mathbf{G}} \end{aligned}$$

# Segregated RK methods

The idea:

1. Consider the projected momentum eq'on on the discretely divergence free space

$$\dot{M\bar{\mathbf{U}}} + P(K + C(\mathbf{U}))\mathbf{U} = P\mathbf{F} + G(DM^{-1}G)^{-1}\dot{\mathbf{G}},$$

with  $P := (I - G(DM^{-1}G)^{-1}DM^{-1})$

# Segregated RK methods

## The idea:

1. Consider the projected momentum eq'on on the discretely divergence free space
2. Integrate the resulting ODE system w/ preferred IMEX RK method (diagonally implicit)...

$$M\dot{\bar{U}} = \mathcal{F}(\bar{U}) + \mathcal{G}(\bar{U}), \quad \mathcal{F} : \text{implicit terms}, \quad \mathcal{G} : \text{explicit ones}$$

# Segregated RK methods

**The idea:**

1. Consider the projected momentum eq'on on the discretely divergence free space
2. Integrate the resulting ODE system w/ preferred IMEX RK method (diagonally implicit)...
3.  $\mathcal{G}$  must include  $G(DM^{-1}G)^{-1} \left( DM^{-1}(K + C(\mathbf{U})\mathbf{U} - \mathbf{F}) + \dot{\mathbf{G}} \right)$  for velocity-pressure segregation

$$M\dot{\mathbf{U}} = \mathcal{F}(\mathbf{U}) + \mathcal{G}(\mathbf{U}), \quad \mathcal{F} : \text{implicit terms}, \quad \mathcal{G} : \text{explicit ones}$$

# Segregated RK methods

## The idea:

1. Consider the projected momentum eq'on on the discretely divergence free space
2. Integrate the resulting ODE system w/ preferred IMEX RK method (diagonally implicit)...
3.  $\mathcal{G}$  must include  $G(DM^{-1}G)^{-1} \left( DM^{-1}(K + C(\mathbf{U})\mathbf{U} - \mathbf{F}) + \dot{\mathbf{G}} \right)$  for velocity-pressure segregation
4. The rest of terms can be treated implicitly / explicitly

$$M\dot{\mathbf{U}} = \mathcal{F}(\mathbf{U}) + \mathcal{G}(\mathbf{U}), \quad \mathcal{F} : \text{implicit terms}, \quad \mathcal{G} : \text{explicit ones}$$

# Segregated RK methods

It leads to the time-discrete algorithm:

$$\frac{1}{\delta t} M \mathbf{U}_i = \frac{1}{\delta t} M \mathbf{U}_n + \sum_{j=1}^i a_{ij} \mathcal{F}(\mathbf{U}_j) + \sum_{j=1}^{i-1} \hat{a}_{ij} \mathcal{G}(\mathbf{U}_j)$$

After re-ordering (pressure again):

$$\begin{aligned} \frac{1}{\delta t} M \mathbf{U}_i &= \frac{1}{\delta t} M \mathbf{U}_n + \sum_{j=1}^i a_{ij} \mathcal{F}(\mathbf{U}_j) + \sum_{j=1}^{i-1} \hat{a}_{ij} \tilde{\mathcal{G}}(\mathbf{U}_j, \mathbf{P}_j), \\ -DM^{-1} G(\mathbf{P}_i) &= DM^{-1}((K + C(\mathbf{U}_i))\mathbf{U}_i - \mathbf{F}(t_i)) + \dot{\mathbf{G}}(t_i) \end{aligned}$$

# Segregated RK methods

It leads to the time-discrete algorithm:

$$\frac{1}{\delta t} M \mathbf{U}_i = \frac{1}{\delta t} M \mathbf{U}_n + \sum_{j=1}^i a_{ij} \mathcal{F}(\mathbf{U}_j) + \sum_{j=1}^{i-1} \hat{a}_{ij} \mathcal{G}(\mathbf{U}_j)$$

After re-ordering (pressure again):

$$\begin{aligned} \frac{1}{\delta t} M \mathbf{U}_i &= \frac{1}{\delta t} M \mathbf{U}_n + \sum_{j=1}^i a_{ij} \mathcal{F}(\mathbf{U}_j) + \sum_{j=1}^{i-1} \hat{a}_{ij} \tilde{\mathcal{G}}(\mathbf{U}_j, \mathbf{P}_j), \\ -DM^{-1} G(\mathbf{P}_i) &= DM^{-1}((K + C(\mathbf{U}_i))\mathbf{U}_i - \mathbf{F}(t_i)) + \dot{\mathbf{G}}(t_i) \end{aligned}$$

- Viscous / convective can be treated implicitly / explicitly

$$\mathcal{F}(\mathbf{U}) := -K\mathbf{U} + \mathbf{F} - C(\mathbf{U})\mathbf{U}, \quad \tilde{\mathcal{G}}(\mathbf{U}, P) = -GP \quad \text{or}$$

$$\mathcal{F}(\mathbf{U}) := -K\mathbf{U} + \mathbf{F}, \quad \tilde{\mathcal{G}}(\mathbf{U}, P) = -C(\mathbf{U})\mathbf{U} - GP$$

# SRK Some properties

- Segregation at the time integration level (no additional splitting)
- High order achievable (the one of the ODE RK integration)
- Implicit LES turbulent model (stabilization terms)
- Equal-order & optimal velocity-pressure methods (of interest in FSI, etc.)
- HERK and SRK methods are *equivalent* for full explicit treatment

# Numerical experiments

Manufactured analytical solutions:

- Simple  $\sin(t) \cdot \exp(t)$  function.

Laminar benchmark:

- 2D Laminar flow around a cylinder.

# Manufactured analytical solution

## Problem setting:

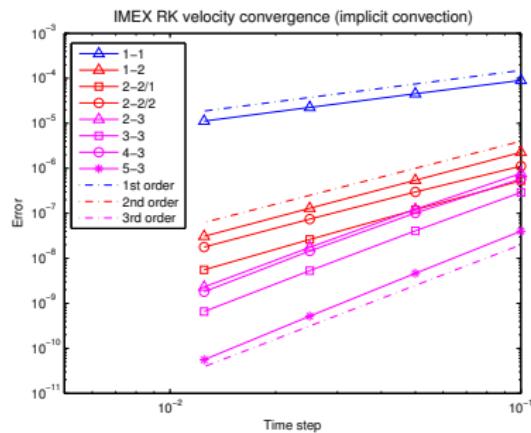
- Analytical solution:

$$\mathbf{u}(x, y, t) = \begin{bmatrix} x \\ -y \end{bmatrix} \sin\left(\frac{\pi}{10}t\right) \exp\left(\frac{t}{25}\right),$$
$$p(x, y) = x + y.$$

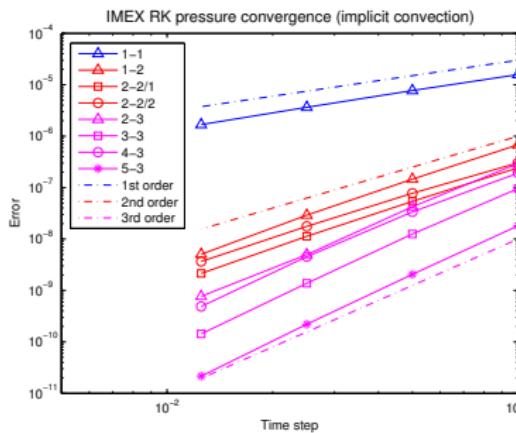
- $Re = 1/10/100$ .
- Different IMEX Butcher tableaus with 1st, 2nd and 3rd order.

# Manufactured analytical solution

## Implicit convection:



(a) Velocity convergence,  $\nu = 0.01$

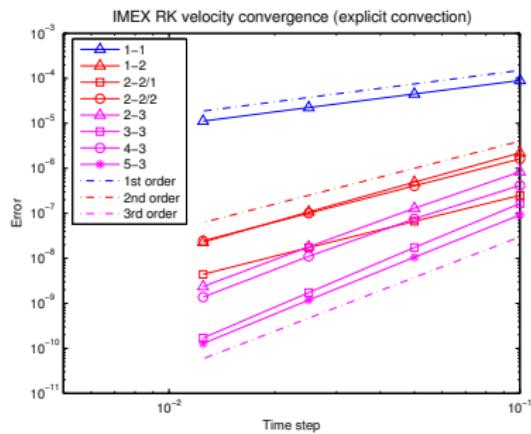


(b) Pressure convergence,  $\nu = 0.01$

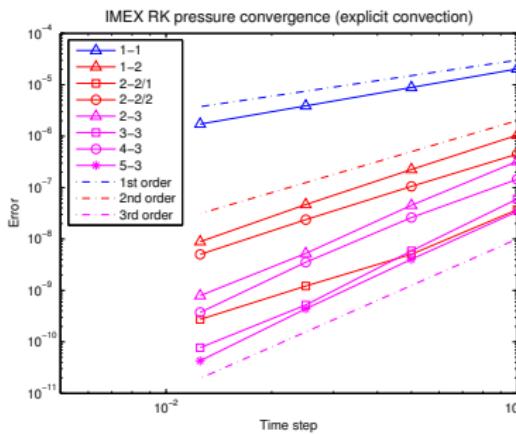
Fully implicit SRK.

# Manufactured analytical solution

## Explicit convection:



(a) Velocity convergence,  $\nu = 0.01$



(b) Pressure convergence,  $\nu = 0.01$

SRK convergence with convection integrated explicitly and diffusion integrated implicitly.

# 2D Laminar flow around a cylinder

## Problem setting:

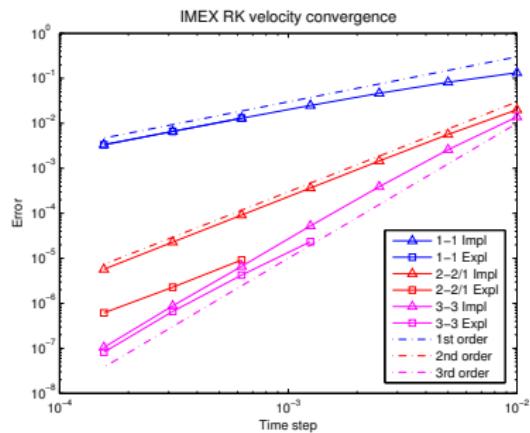
- Widely used benchmark.
- $Re = 100$ .
- Time convergence.
- Drag and Lift coefficients.



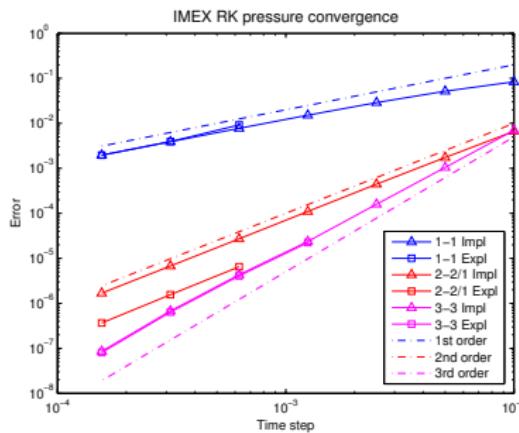
Vorticity field at  $t = 8.0$ .

# 2D Laminar flow around a cylinder

## Implicit convection:



(a) Velocity convergence



(b) Pressure convergence

Fully implicit and IMEX-SRK convergence rate comparison.

# Segregated Runge-Kutta Conclusions

- Velocity-pressure segregated by the time integrator (IMEX RK)

# Segregated Runge-Kutta Conclusions

- Velocity-pressure segregated by the time integrator (IMEX RK)
- No additional splitting needed

# Segregated Runge-Kutta Conclusions

- Velocity-pressure segregated by the time integrator (IMEX RK)
- No additional splitting needed
- Equal order in time (pressure error not spoiled)

# Segregated Runge-Kutta Limitations

- Only for index-2 DAE.

# Segregated Runge-Kutta Limitations

- Only for index-2 DAE.
- Not applicable for equal-order stabilization methods.

## 1. Motivation

## 2. Residual-based VMS

## 3. Mixed FE VMS

## 4. Segregated Runge-Kutta

## 5. Segregated VMS

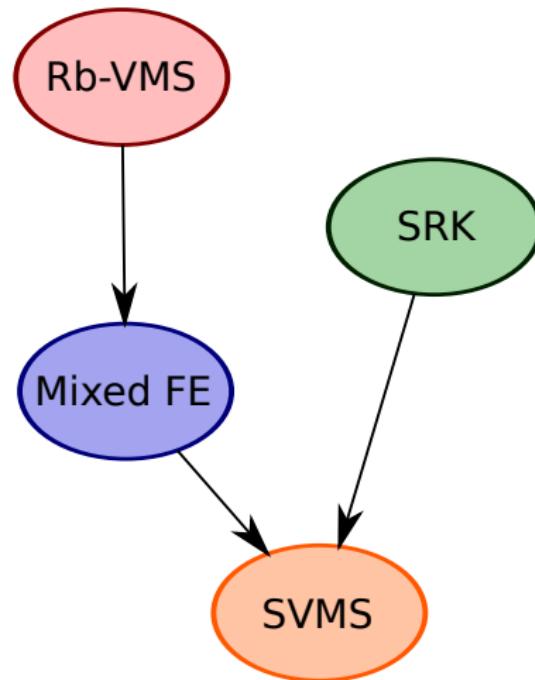
Formulation

Large-scale solvers

Numerical experiments

Conclusions

## 6. Conclusions



# Motivation

## Step by step...

- Residual-based VMS as LES models.
- Mixed FE formulations LES.
- High-order FE methods.
- High-order time integration schemes.
- Adaptive time stepping techniques.
- Velocity-pressure segregation.
- Scalable solvers.
- Application.

# Looking backward...

## VMS as LES models: Residual-based VMS methods

- Velocity + pressure stabilization, applied to equal-order FEs.
- Intensively tested methods (several works in the literature).
- Nature of the problem changes, not DAE-2 type (!).
- It prevents us to use segregated RK methods.

# Looking backward...

## VMS as LES models: Mixed FE VMS methods

- Only the term that we need, i.e., convection stabilization.
- To keep accuracy, we use orthogonal projections ( $\mathcal{P}_h^\perp$ ).
- Many choices for the  $\mathcal{P}_h$  projector: local, global (OSS).
- Discrete problem still index-2 DAE.
- Segregated RK schemes can be used.

# Looking backward...

## VMS as LES models: Mixed FE VMS methods

- Only the term that we need, i.e., convection stabilization.
- To keep accuracy, we use orthogonal projections ( $\mathcal{P}_h^\perp$ ).
- Many choices for the  $\mathcal{P}_h$  projector: local, global (OSS).
- Discrete problem still index-2 DAE.
- Segregated RK schemes can be used.

Now, we can use Segregated RK methods for ILES of turbulent flows

# Formulation

**OSS-ISS:** Semi-discrete form

$$\begin{bmatrix} M\dot{\mathbf{U}} \\ \mathbf{0} \\ \mathbf{0} \end{bmatrix} + \begin{bmatrix} K + C(\mathbf{U}) + A_\tau & G & B_\tau \\ D & 0 & 0 \\ -B_\tau^T & 0 & M_\tau \end{bmatrix} \begin{bmatrix} \mathbf{U} \\ \mathbf{P} \\ \boldsymbol{\Upsilon} \end{bmatrix} = \begin{bmatrix} \mathbf{F}_u \\ \mathbf{0} \\ \mathbf{0} \end{bmatrix},$$

# Formulation

OSS-ISS: Semi-discrete form

$$\begin{aligned} M\dot{\mathbf{U}} + (K + C(\mathbf{U}) + A_\tau)\mathbf{U} + G\mathbf{P} + B_\tau\boldsymbol{\Upsilon} &= \mathbf{F}_u, \\ M_\tau\boldsymbol{\Upsilon} - B_\tau^T\mathbf{U} &= \mathbf{0}, \\ D\mathbf{U} &= \mathbf{0}. \end{aligned}$$

# Formulation

## OSS-ISS:

1. Consider the projected momentum eq'on on the discretely divergence free space

$$\dot{M\mathbf{U}} + (K + C(\mathbf{U}) + A_\tau)\mathbf{U} + G\mathbf{P} + B_\tau \boldsymbol{\Upsilon} = \mathbf{F}_u,$$

$$M_\tau \boldsymbol{\Upsilon} - B_\tau^T \mathbf{U} = \mathbf{0},$$

$$-DM^{-1}G\mathbf{P} = DM^{-1}(K + C(\mathbf{U})\mathbf{U} + B_\tau \boldsymbol{\Upsilon} - \mathbf{F}_u).$$

# Formulation

## OSS-ISS:

1. Consider the projected momentum eq'on on the discretely divergence free space
2. Integrate the resulting ODE system w/ preferred IMEX RK method (diagonally implicit)...

$$\dot{M\mathbf{U}} = \mathcal{F}(\mathbf{U}, \boldsymbol{\Upsilon}) + \mathcal{G}(\mathbf{U}, \mathbf{P}), \quad \mathcal{F} : \text{implicit terms, } \mathcal{G} : \text{explicit ones}$$

$$M_\tau \boldsymbol{\Upsilon} - B_\tau^T \mathbf{U} = \mathbf{0},$$

$$-DM^{-1}\mathbf{GP} = DM^{-1}((K + C(\mathbf{U}) + A_\tau)\mathbf{U} + B_\tau \boldsymbol{\Upsilon} - \mathbf{F}_u).$$

# Formulation

## OSS-ISS:

1. Consider the projected momentum eq'on on the discretely divergence free space
2. Integrate the resulting ODE system w/ preferred IMEX RK method (diagonally implicit)...
3.  $\mathcal{G}$  must include  

$$G(DM^{-1}G)^{-1} \left( DM^{-1}(K + C(\mathbf{U}) + A_\tau)\mathbf{U} + B_\tau \boldsymbol{\Upsilon} - \mathbf{F}_u \right)$$
 for velocity-pressure segregation

$$\dot{M\mathbf{U}} = \mathcal{F}(\mathbf{U}, \boldsymbol{\Upsilon}) + \mathcal{G}(\mathbf{U}, \mathbf{P}), \quad \mathcal{F} : \text{implicit terms, } \mathcal{G} : \text{explicit ones}$$

$$M_\tau \boldsymbol{\Upsilon} - B_\tau^T \mathbf{U} = \mathbf{0},$$

$$-DM^{-1}\mathbf{GP} = DM^{-1}((K + C(\mathbf{U}) + A_\tau)\mathbf{U} + B_\tau \boldsymbol{\Upsilon} - \mathbf{F}_u).$$

# Formulation

## OSS-ISS:

1. Consider the projected momentum eq'on on the discretely divergence free space
2. Integrate the resulting ODE system w/ preferred IMEX RK method (diagonally implicit)...
3.  $\mathcal{G}$  must include  

$$G(DM^{-1}G)^{-1} \left( DM^{-1}(K + C(\mathbf{U}) + A_\tau)\mathbf{U} + B_\tau \boldsymbol{\Upsilon} - \mathbf{F}_u \right)$$
 for velocity-pressure segregation
4. We treat the projection term implicit or explicit, according the convective term.

$$\dot{M\mathbf{U}} = \mathcal{F}(\mathbf{U}, \boldsymbol{\Upsilon}) + \mathcal{G}(\mathbf{U}, \mathbf{P}), \quad \mathcal{F} : \text{implicit terms, } \mathcal{G} : \text{explicit ones}$$

$$M_\tau \boldsymbol{\Upsilon} - B_\tau^T \mathbf{U} = \mathbf{0},$$

$$-DM^{-1}\mathbf{GP} = DM^{-1}((K + C(\mathbf{U}) + A_\tau)\mathbf{U} + B_\tau \boldsymbol{\Upsilon} - \mathbf{F}_u).$$

# Formulation

## OSS-ISS:

1. Consider the projected momentum eq'on on the discretely divergence free space
2. Integrate the resulting ODE system w/ preferred IMEX RK method (diagonally implicit)...
3.  $\mathcal{G}$  must include  

$$G(DM^{-1}G)^{-1} \left( DM^{-1}(K + C(\mathbf{U}) + A_\tau)\mathbf{U} + B_\tau \boldsymbol{\Upsilon} - \mathbf{F}_u \right)$$
 for velocity-pressure segregation
4. We treat the projection term implicit or explicit, according the convective term.

$$\dot{M\mathbf{U}} = \mathcal{F}(\mathbf{U}) + \mathcal{G}(\mathbf{U}, \boldsymbol{\Upsilon}, \mathbf{P}), \quad \mathcal{F} : \text{implicit terms, } \mathcal{G} : \text{explicit ones}$$

$$M_\tau \boldsymbol{\Upsilon} - B_\tau^T \mathbf{U} = \mathbf{0},$$

$$-DM^{-1}\mathbf{GP} = DM^{-1}((K + C(\mathbf{U}) + A_\tau)\mathbf{U} + B_\tau \boldsymbol{\Upsilon} - \mathbf{F}_u).$$

## Final discrete problem

At each stage of the Runge-Kutta scheme: (explicit convective term and projection version)

$$\frac{1}{\delta t} M \mathbf{U}_i = \frac{1}{\delta t} M \mathbf{U}_n + \sum_{j=1}^i a_{ij} \mathcal{F}(\mathbf{U}_j) + \sum_{j=1}^{i-1} \hat{a}_{ij} \mathcal{G}(\mathbf{U}_j, \boldsymbol{\Upsilon}_j, \mathbf{P}_j),$$

$$M_\tau \boldsymbol{\Upsilon}_i - B_\tau^T \mathbf{U}_i = \mathbf{0},$$

$$- DM^{-1} G(\mathbf{P}_i) = DM^{-1} ((K + C(\mathbf{U}_i) + A_\tau) \mathbf{U}_i + B_\tau \boldsymbol{\Upsilon}_i - \mathbf{F}_u(t_i)).$$

## Final discrete problem

At each stage of the Runge-Kutta scheme: (explicit convective term and projection version)

$$\left( \frac{1}{\delta t} M + a_{ii} K \right) \mathbf{U}_i = \frac{1}{\delta t} M \mathbf{U}_n + \sum_{j=1}^{i-1} [a_{ij} \mathcal{F}(\mathbf{U}_j) + \hat{a}_{ij} \mathcal{G}(\mathbf{U}_j, \boldsymbol{\Upsilon}_j, \mathbf{P}_j)],$$

$$M_\tau \boldsymbol{\Upsilon}_i - B_\tau^T \mathbf{U}_i = \mathbf{0},$$

$$- DM^{-1} G(\mathbf{P}_i) = DM^{-1} ((K + C(\mathbf{U}_i) + A_\tau) \mathbf{U}_i + B_\tau \boldsymbol{\Upsilon}_i - \mathbf{F}_u(t_i)).$$

## Final discrete problem

At each stage of the Runge-Kutta scheme: (explicit convective term and projection version)

$$\left( \frac{1}{\delta t} M + a_{ii} K \right) \mathbf{U}_i = \frac{1}{\delta t} M \mathbf{U}_n + \sum_{j=1}^{i-1} [a_{ij} \mathcal{F}(\mathbf{U}_j) + \hat{a}_{ij} \mathcal{G}(\mathbf{U}_j, \boldsymbol{\Upsilon}_j, \mathbf{P}_j)],$$

$$M_\tau \boldsymbol{\Upsilon}_i - B_\tau^T \mathbf{U}_i = \mathbf{0},$$

$$- DM^{-1} G(\mathbf{P}_i) = DM^{-1} ((K + C(\mathbf{U}_i) + A_\tau) \mathbf{U}_i + B_\tau \boldsymbol{\Upsilon}_i - \mathbf{F}_u(t_i)).$$

- **Elasticity-type** matrix for the momentum equation.

## Final discrete problem

At each stage of the Runge-Kutta scheme: (explicit convective term and projection version)

$$\left( \frac{1}{\delta t} M + a_{ii} K \right) \mathbf{U}_i = \frac{1}{\delta t} M \mathbf{U}_n + \sum_{j=1}^{i-1} [a_{ij} \mathcal{F}(\mathbf{U}_j) + \hat{a}_{ij} \mathcal{G}(\mathbf{U}_j, \boldsymbol{\Upsilon}_j, \mathbf{P}_j)],$$

$$\textcolor{red}{M_\tau \boldsymbol{\Upsilon}_i - B_\tau^T \mathbf{U}_i = \mathbf{0}},$$

$$- DM^{-1} G(\mathbf{P}_i) = DM^{-1} ((K + C(\mathbf{U}_i) + A_\tau) \mathbf{U}_i + B_\tau \boldsymbol{\Upsilon}_i - \mathbf{F}_u(t_i)).$$

- Elasticity-type matrix for the momentum equation.
- **Mass-type** matrix for the projection equation.

## Final discrete problem

At each stage of the Runge-Kutta scheme: (explicit convective term and projection version)

$$\left( \frac{1}{\delta t} M + a_{ii} K \right) \mathbf{U}_i = \frac{1}{\delta t} M \mathbf{U}_n + \sum_{j=1}^{i-1} [a_{ij} \mathcal{F}(\mathbf{U}_j) + \hat{a}_{ij} \mathcal{G}(\mathbf{U}_j, \boldsymbol{\Upsilon}_j, \mathbf{P}_j)],$$

$$M_\tau \boldsymbol{\Upsilon}_i - B_\tau^T \mathbf{U}_i = \mathbf{0},$$

$$- \textcolor{red}{DM^{-1} G(\mathbf{P}_i)} = DM^{-1} ((K + C(\mathbf{U}_i) + A_\tau) \mathbf{U}_i + B_\tau \boldsymbol{\Upsilon}_i - \mathbf{F}_u(t_i)).$$

- Elasticity-type matrix for the momentum equation.
- Mass-type matrix for the projection equation.
- **Darcy-type** system for the pressure equation.

# Large scale simulations

**Momentum system matrix:**  $\frac{1}{\delta t}M + a_{ii}K$

- Implicit viscous: Balancing domain decomposition (BDDC) for  $M + \delta t K$
- Coercive operator, algorithmically scalable algorithm keeping constant the local problem size

# Large scale simulations

**Momentum system matrix:**  $\frac{1}{\delta t}M + a_{ii}K$

- Implicit viscous: Balancing domain decomposition (BDDC) for  $M + \delta t K$
- Coercive operator, algorithmically scalable algorithm keeping constant the local problem size

**Pressure system matrix:**  $DM^{-1}G$

- Spectrally equivalent to standard Laplacian  $L_p$
- Written as

$$\begin{bmatrix} M & G \\ D & 0 \end{bmatrix} \quad \text{preconditioned w/} \quad \begin{bmatrix} M & 0 \\ 0 & L_{\text{BDDC}} \end{bmatrix}$$

- Optimal + scalable method (BDDC)

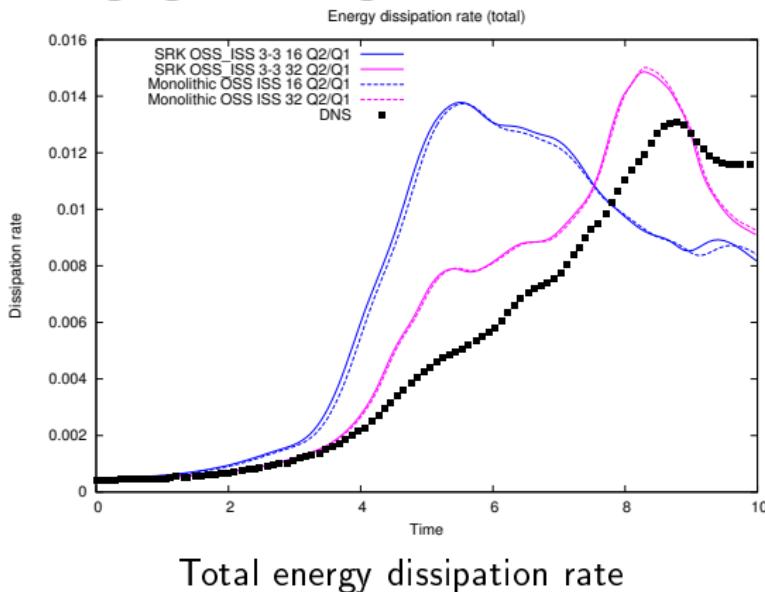
# Numerical experiments

Three different turbulent benchmarks:

- Taylor-Green Vortex (TGV) flow.
- Turbulent Channel Flow (TCF).
- Flow around a NACA profile.

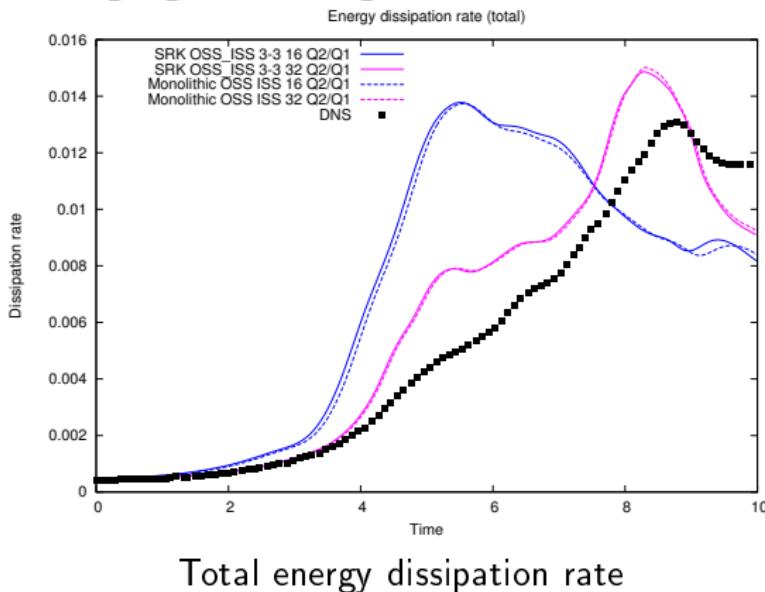
# TGV Taylor-Green Vortex flow

## Monolithic vs Segregated Runge-Kutta:



# TGV Taylor-Green Vortex flow

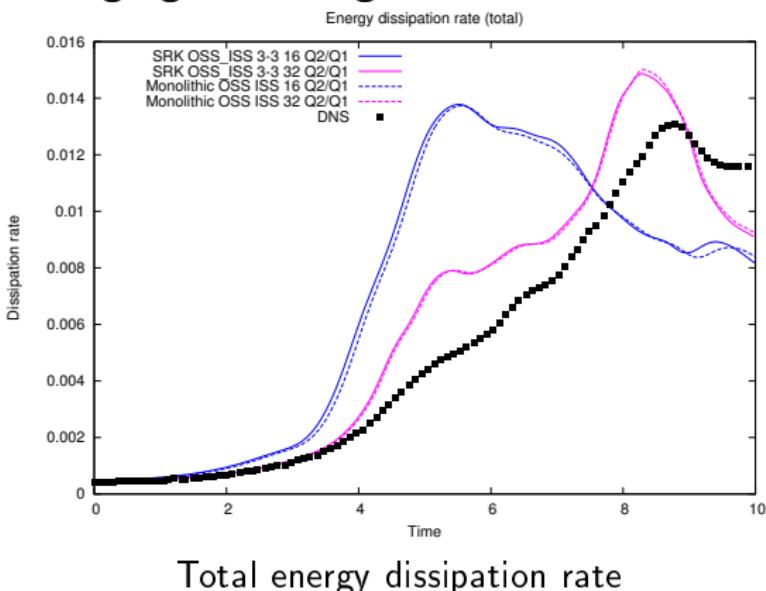
## Monolithic vs Segregated Runge-Kutta:



- Almost **Identical results** obtained with Crank-Nicolson.

# TGV Taylor-Green Vortex flow

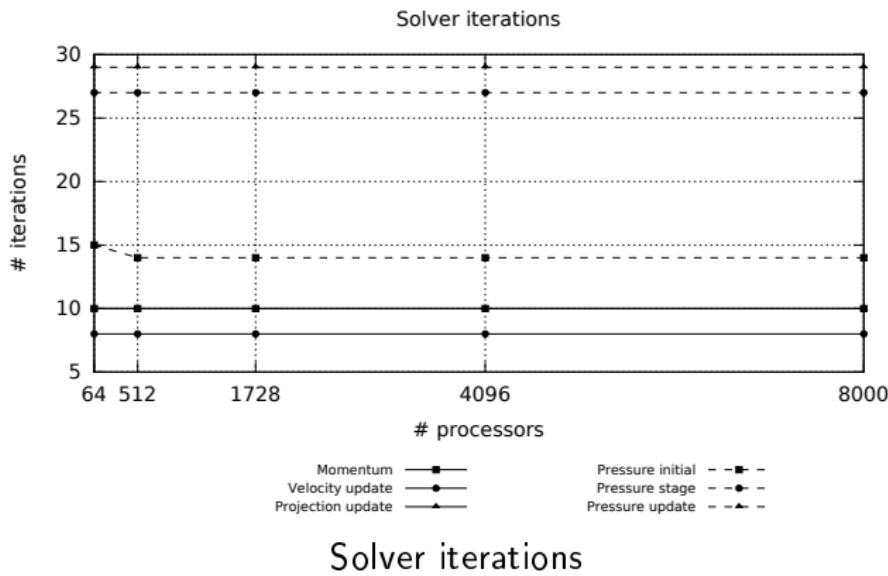
## Monolithic vs Segregated Runge-Kutta:



- Almost Identical results obtained with Crank-Nicolson.
- Velocity-pressure segregation and adaptive time stepping.

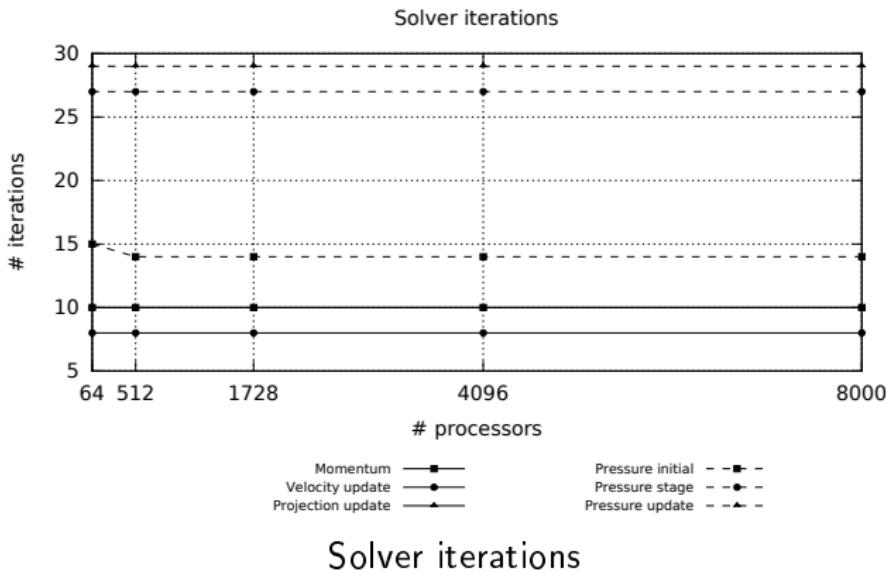
# TGV Taylor-Green Vortex flow

**Weak scalability:** HLRN-III Cray XC40 with Intel Xeon Haswell



# TGV Taylor-Green Vortex flow

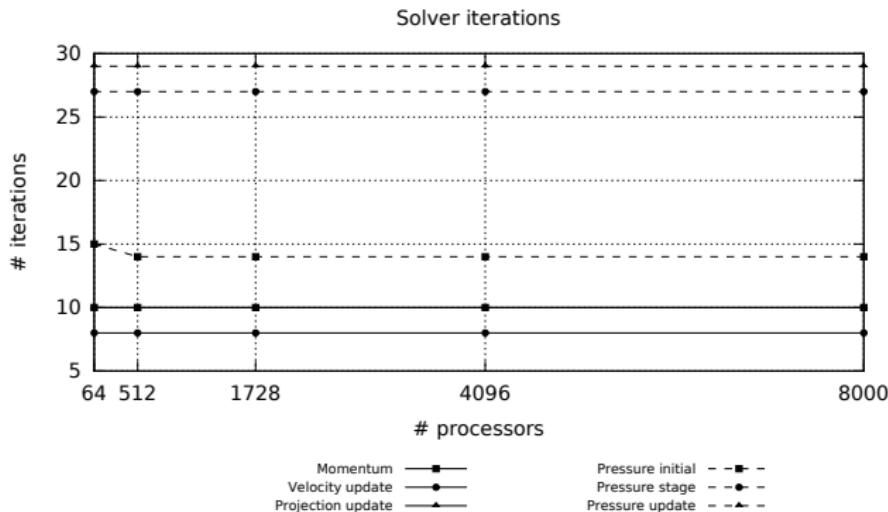
**Weak scalability:** HLRN-III Cray XC40 with Intel Xeon Haswell



- Perfect scalability in terms of solver iterations.

# TGV Taylor-Green Vortex flow

**Weak scalability:** HLRN-III Cray XC40 with Intel Xeon Haswell

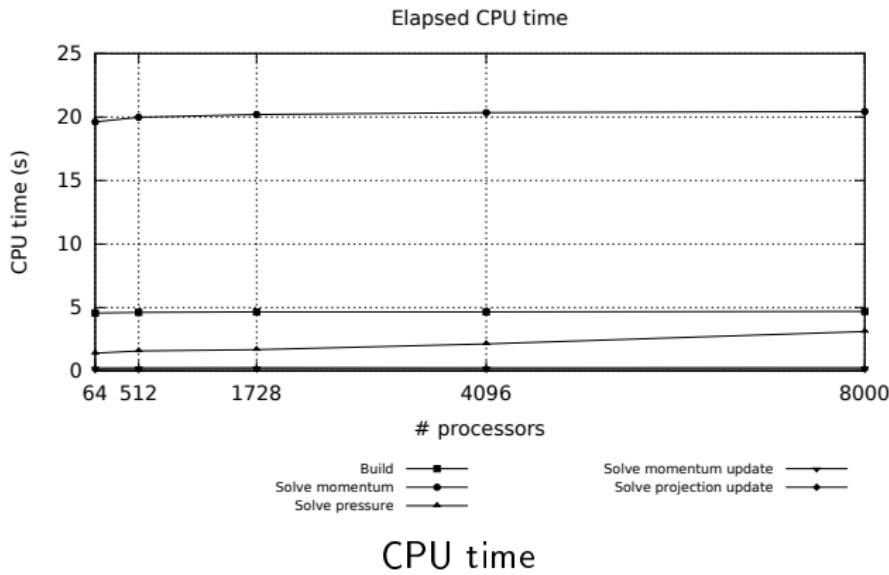


### Solver iterations

- Perfect scalability in terms of solver iterations.
- Darcy-type problem requires **more iterations**.

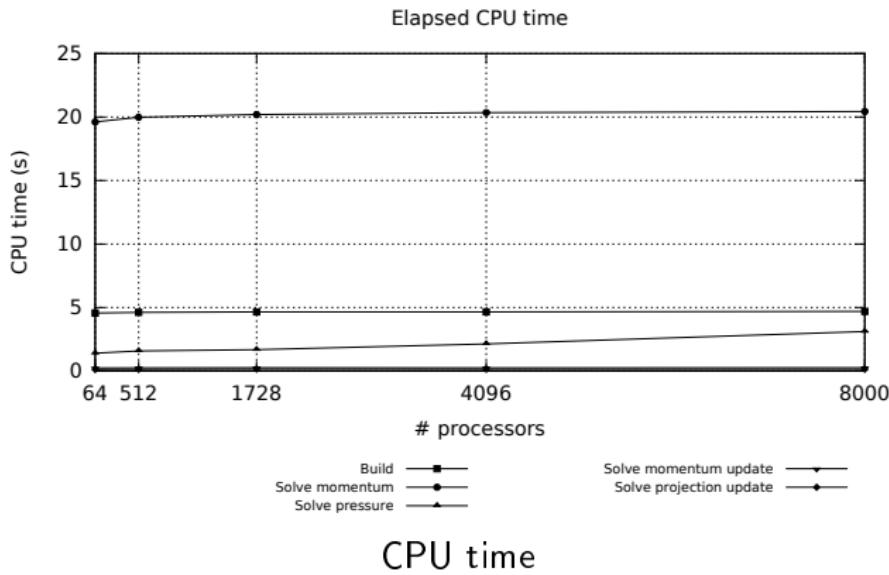
# TGV Taylor-Green Vortex flow

**Weak scalability:** HLRN-III Cray XC40 with Intel Xeon Haswell



# TGV Taylor-Green Vortex flow

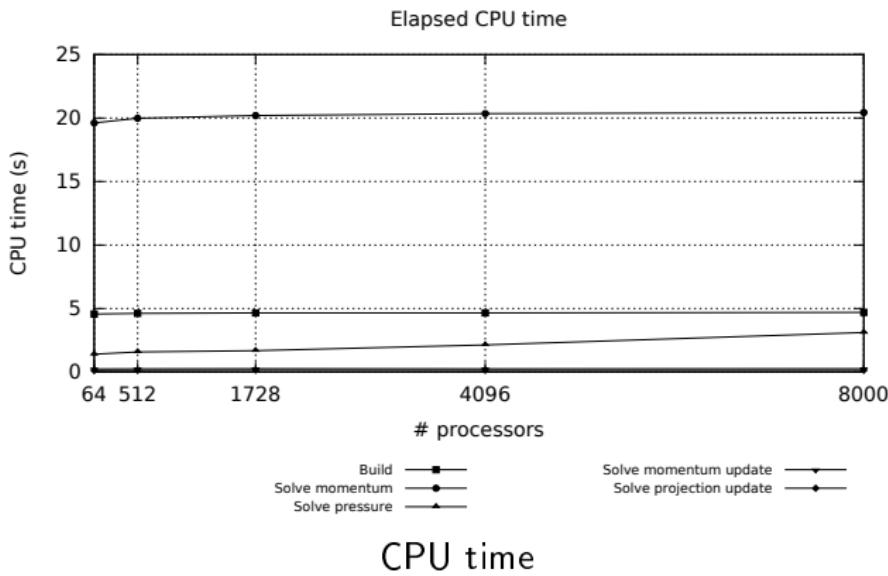
**Weak scalability:** HLRN-III Cray XC40 with Intel Xeon Haswell



- Perfect scalability in CPU time consumed.

# TGV Taylor-Green Vortex flow

**Weak scalability:** HLRN-III Cray XC40 with Intel Xeon Haswell



- Perfect scalability in CPU time consumed.
- Momentum problem requires more time.

# TCF Turbulent Channel Flow at $Re_\tau = 395$

## LES for wall bounded flows?

### Problem:

- LES models cannot accurately simulate boundary layers without stretched meshes

# TCF Turbulent Channel Flow at $Re_\tau = 395$

## LES for wall bounded flows?

### Problem:

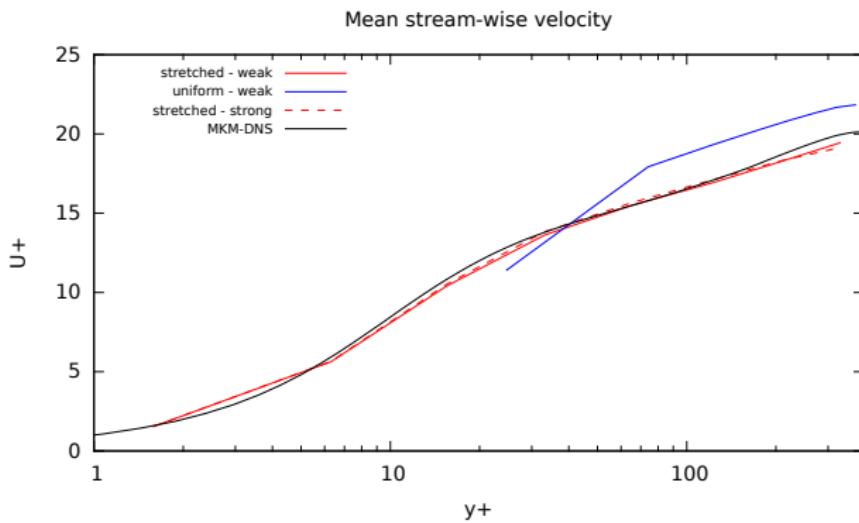
- LES models cannot accurately simulate boundary layers without stretched meshes

### Solution:

- Use of weak boundary conditions (Nitche's method)
- Impose the tangential traction on the wall from a wall law model [Bazilevs et al '07]
- We also impose weakly the wall-normal component

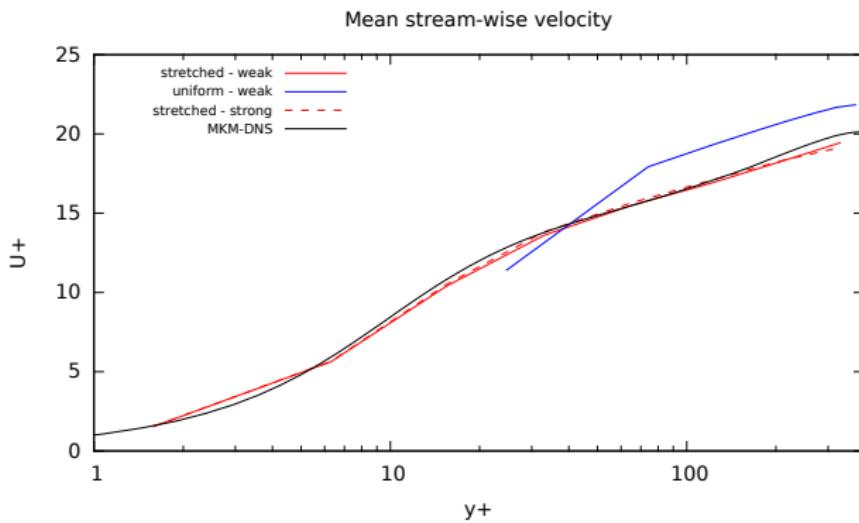
# TCF Turbulent Channel Flow at $Re_\tau = 395$

Weak boundary conditions:



# TCF Turbulent Channel Flow at $Re_\tau = 395$

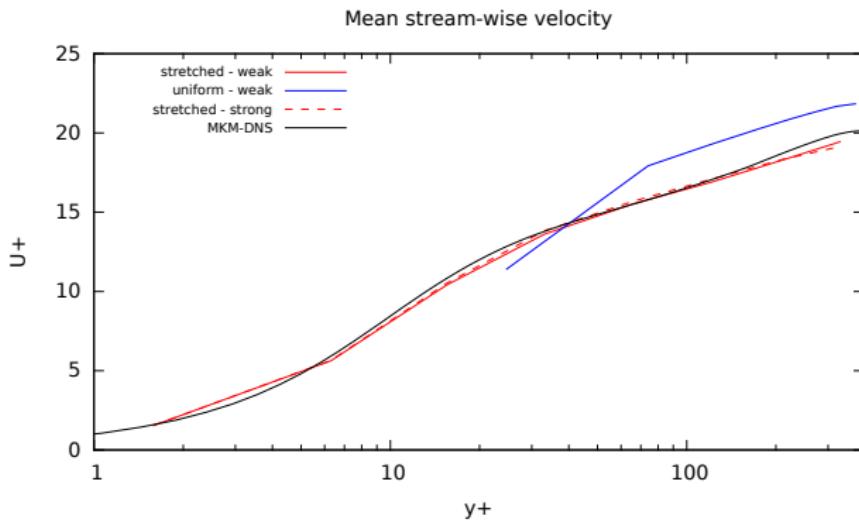
Weak boundary conditions:



- We recover the strong BC results when stretched mesh is used.

# TCF Turbulent Channel Flow at $Re_\tau = 395$

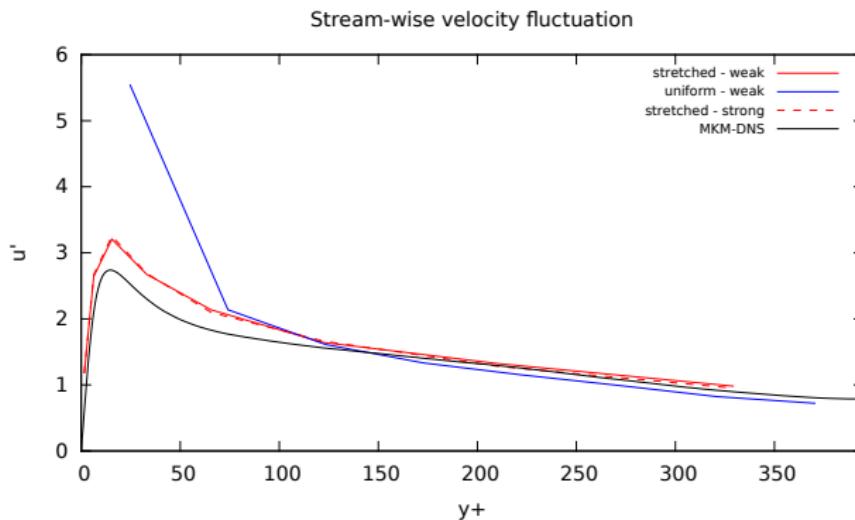
**Weak boundary conditions:**



- We recover the strong BC results when stretched mesh is used.
- **Acceptable results for uniform meshes.**

# TCF Turbulent Channel Flow at $Re_\tau = 395$

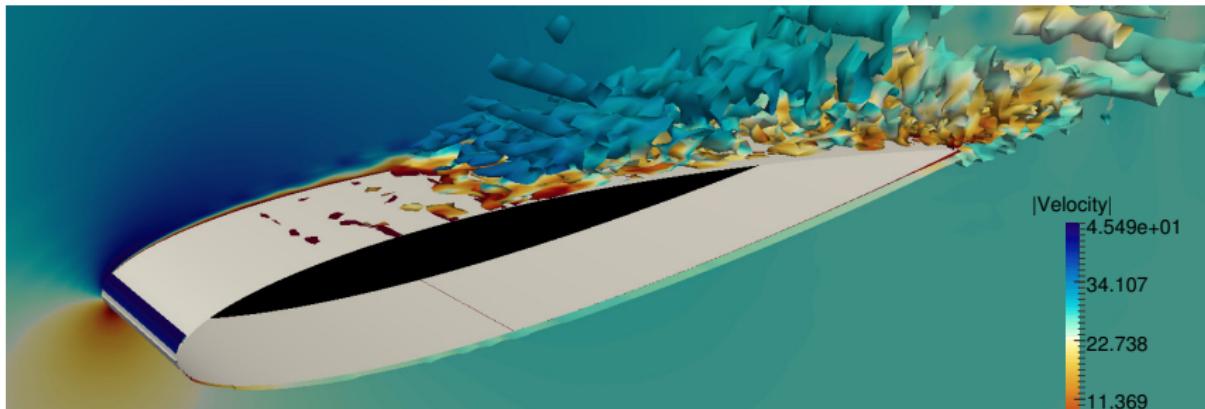
Weak boundary conditions:



- We recover the strong BC results when stretched mesh is used.
- Acceptable results for uniform meshes.

# Turbulent flow around a NACA profile

- Low Reynolds number,  $Re = 23000$ .
- Weak boundary conditions **also on the wall-normal component**.
- Coarse mesh.



Q-criterion isosurfaces for a fully developed flow

# Segregated VMS Conclusions

- Implicit LES for mixed FEs + convection stabilization
  - Similar behavior as VMS-type solvers
  - Not affecting nature of the system (index-2 DAE)
- Large scale computations
  - Segregation leads to coercive blocks to be solved
  - Highly scalable multilevel/overlapped implementation of BDDC
- Weak boundary conditions on all components

## 1. Motivation

## 2. Residual-based VMS

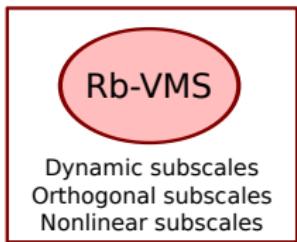
## 3. Mixed FE VMS

## 4. Segregated Runge-Kutta

## 5. Segregated VMS

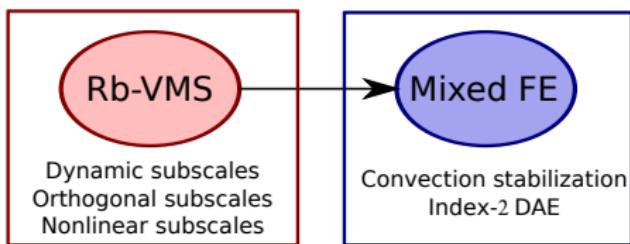
## 6. Conclusions

# Final Conclusions



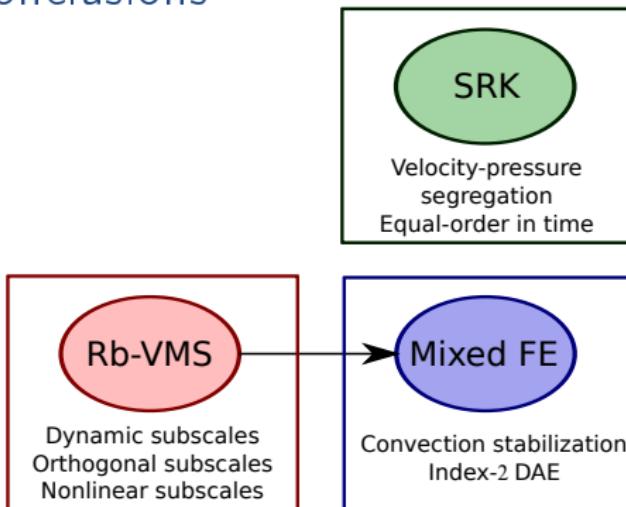
- Demonstrated suitability of Residual-based VMS methods as LES models

# Final Conclusions



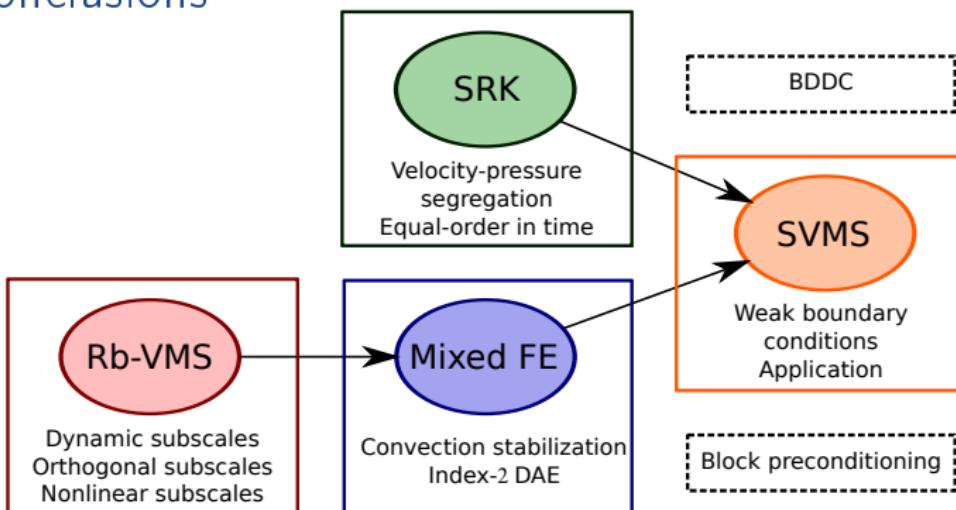
- Demonstrated suitability of Residual-based VMS methods as LES models
- Defined **Mixed FE methods with convection stabilization**

# Final Conclusions



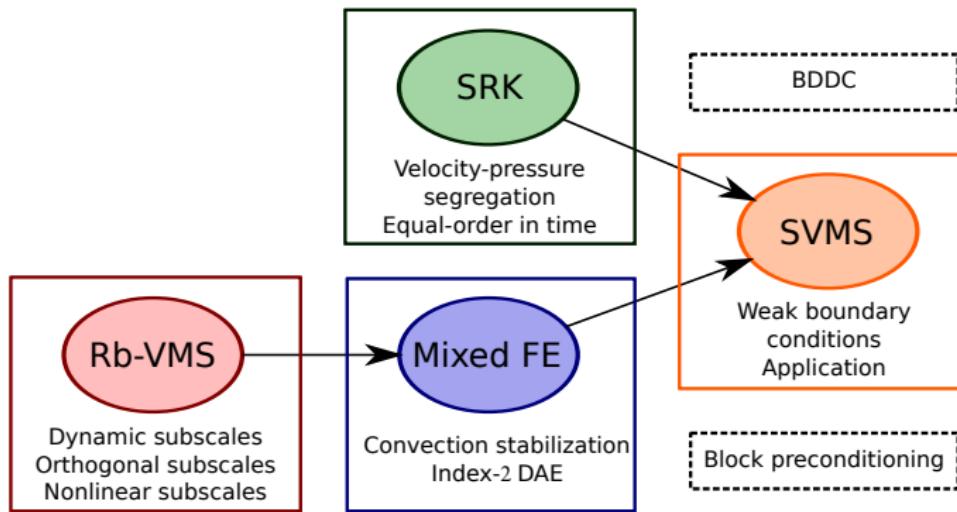
- Demonstrated suitability of Residual-based VMS methods as LES models
- Defined Mixed FE methods with convection stabilization
- Developed novel **Segregated Runge-Kutta methods**

# Final Conclusions



- Demonstrated suitability of Residual-based VMS methods as LES models
- Defined Mixed FE methods with convection stabilization
- Developed novel Segregated Runge-Kutta methods
- **Highly-scalable FE solvers** for the simulation of turbulent incompressible flows

# Final Conclusions



## Thesis goal

Highly scalable Finite Element (FE) framework for Large Eddy Simulations (LES) of incompressible turbulent flows

# Thank you!