Incompact3d User Group Meeting

Some problems to fix and further challenges

Eric Lamballais/Sylvain Laizet

Outline

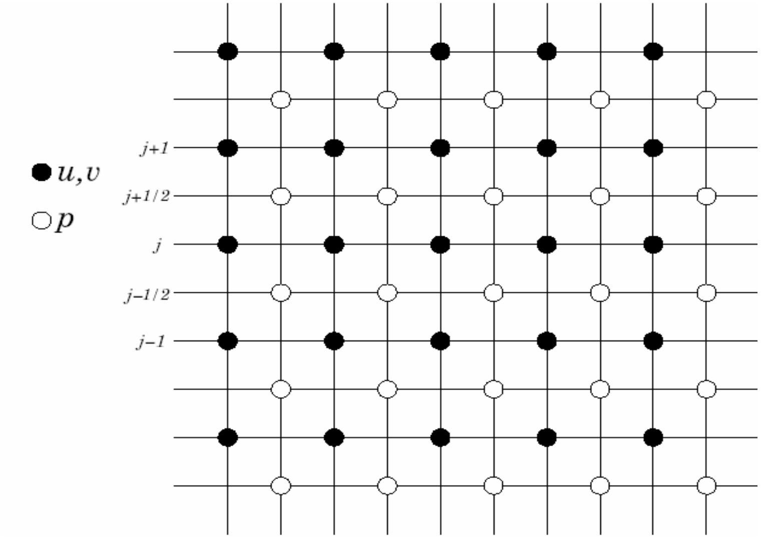
-Some problems to fix

- ✓Poisson solver: quasi-singular modes
- ✓Boundary conditions for pressure (with or without IBM)
- ✓Mass conservation at marginal resolution with IBM

-Further challenges

- ✓Hybrid approach for exascale supercomputers
- ✓Free surface
- ✓Multiblock domain
- ✓Stretching in 2 directions
- ✓2D version of Incompact3d
- ✓Quasicompact3d and Compact3d
- ✓How to deal with users + their developments

Collocated or staggered mesh



✓ Collocated mesh for convective and diffusive terms

✓ Staggered mesh for the pressure treatment

First derivative on a collocated mesh

$$\alpha f'_{i-1} + f'_i + \alpha f'_{i+1} = a\,\frac{f_{i+1} - f_{i-1}}{2\Delta x} + b\,\frac{f_{i+2} - f_{i-2}}{4\Delta x}$$

First derivative on a staggered mesh

$$\alpha f'_{i-1/2} + f'_{i+1/2} + \alpha f'_{i+3/2} = a\,\frac{f_{i+1} - f_i}{\Delta x} + b\,\frac{f_{i+2} - f_{i-1}}{3\Delta x}$$

Mid-point interpolation

$$\alpha f^I_{i-1/2} + f^I_{i+1/2} + \alpha f^I_{i+3/2} = a\,\frac{f_{i+1} - f_i}{2} + b\,\frac{f_{i+2} - f_{i-1}}{2}$$

Poisson solving stage

Using a generic 3D FFT for the pressure

$$\hat{p}_{lmn} = \frac{1}{n_x n_y n_z} \sum_i \sum_j \sum_k p_{ijk} W_x(k_x x_i) W_y(k_y y_j) W_z(k_z z_k)$$

the solving of the Poisson equation consists in

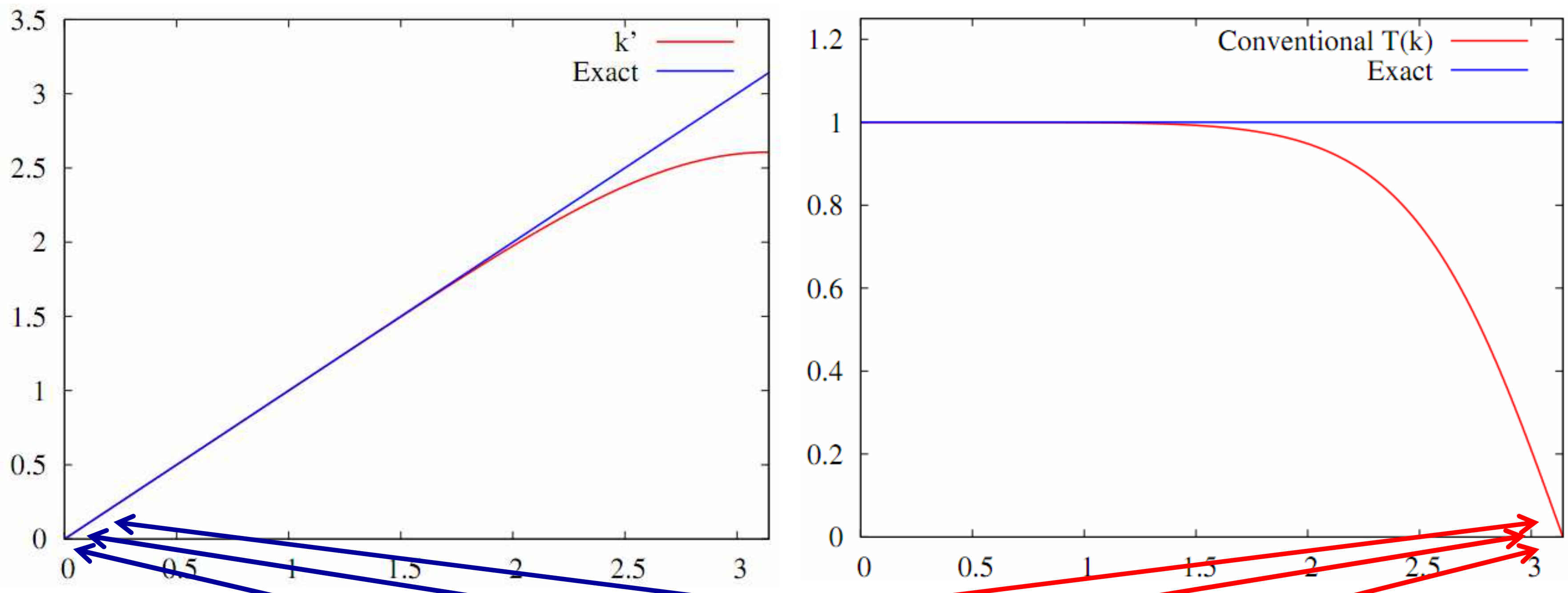
$$\hat{\tilde{p}}^{k+1}_{lmn} = \frac{\widehat{D}_{lmn}}{F_{lmn}}$$ with

$$F_{lmn} = -[(k'_x T_y T_z)^2 + (k'_y T_x T_z)^2 + (k'_z T_x T_y)^2] c_k \Delta t$$

If F_{lmn}=0, no problem (can be ignored while $\nabla . \mathbf{u}^{k+1} = 0$)
If F_{lmn}≈0, potential problem (cannot be ignored)

→ problem with "quasi-singular" modes

Where are quasi-singular modes?

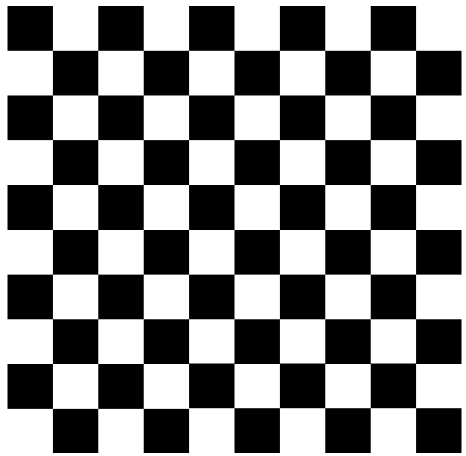


$$F_{lmn} = -[(k'_x T_y T_z)^2 + (k'_y T_x T_z)^2 + (k'_z T_x T_y)^2] c_k \Delta t$$

Example :

$T_x \approx 0$ ($k_x \approx \pi$), $T_y \approx 0$ ($k_y \approx \pi$)

→ quasi-checkerboard mode



Practical consequences of quasi-singular modes

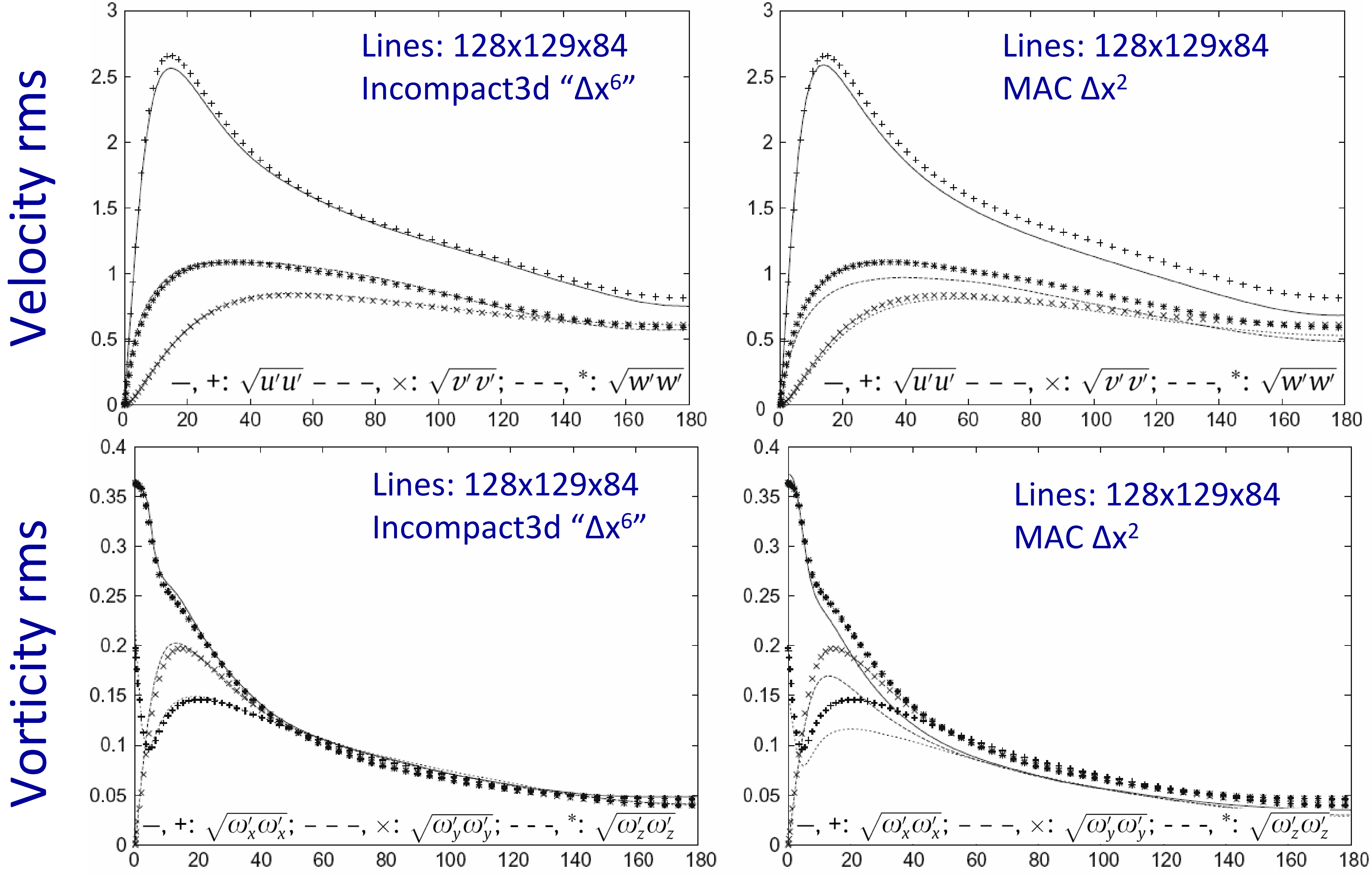
- For well resolved DNS, no problem
  (no oscillation to amplify)
- For marginally resolved DNS, aliasing errors are amplified by quasi-singular modes and Incompact3d behaves less favourably than a conventional second order code

  → high-order numerical dissipation can control aliasing errors while restoring the superiority of Incompact3d
- For marginally resolved LES, numerical dissipation is not enough

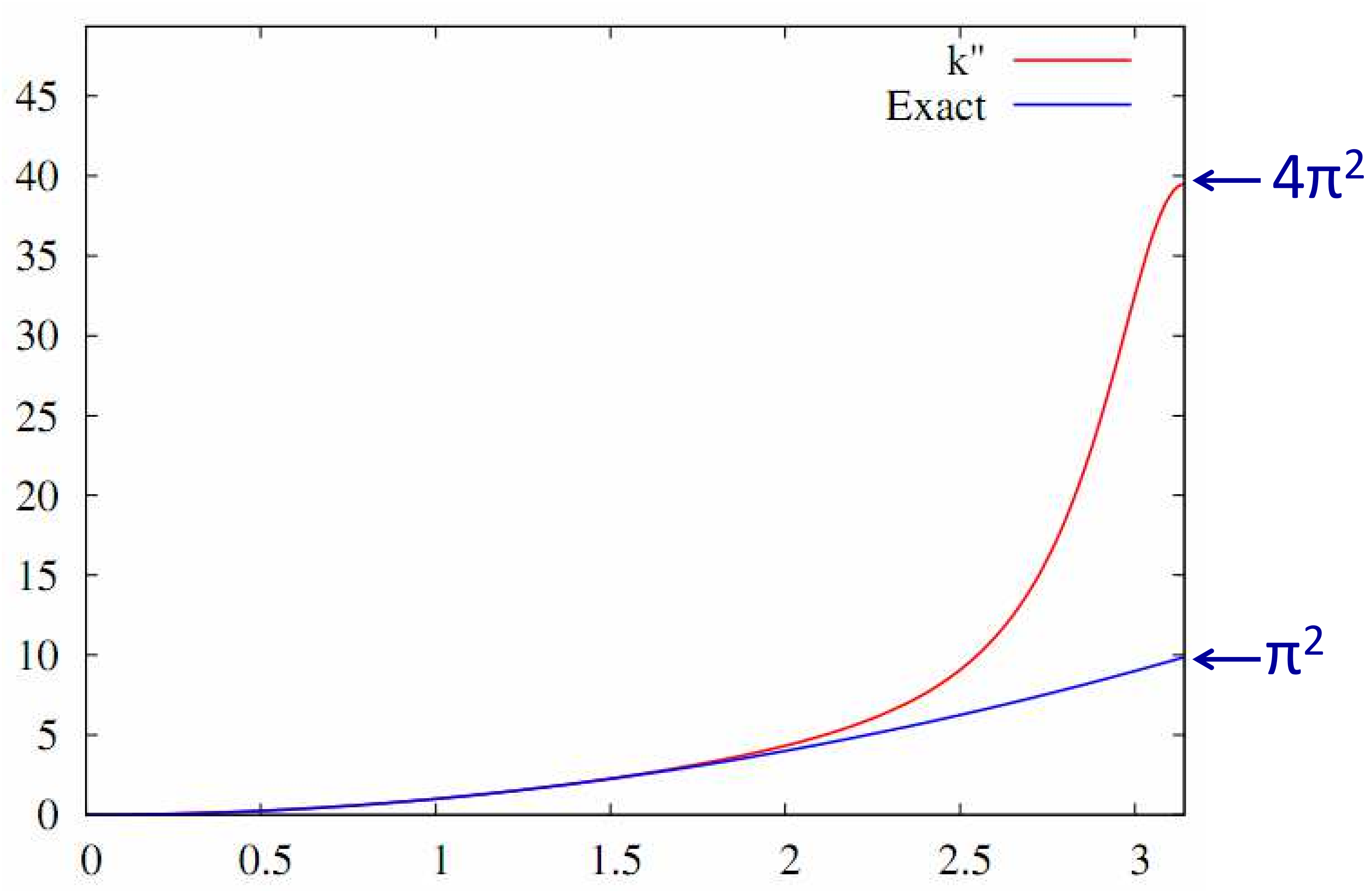
  → unconventional interpolator clearly improves turbulent statistics but a residual zigzag pattern can be identified

Example of DNS at marginal resolution

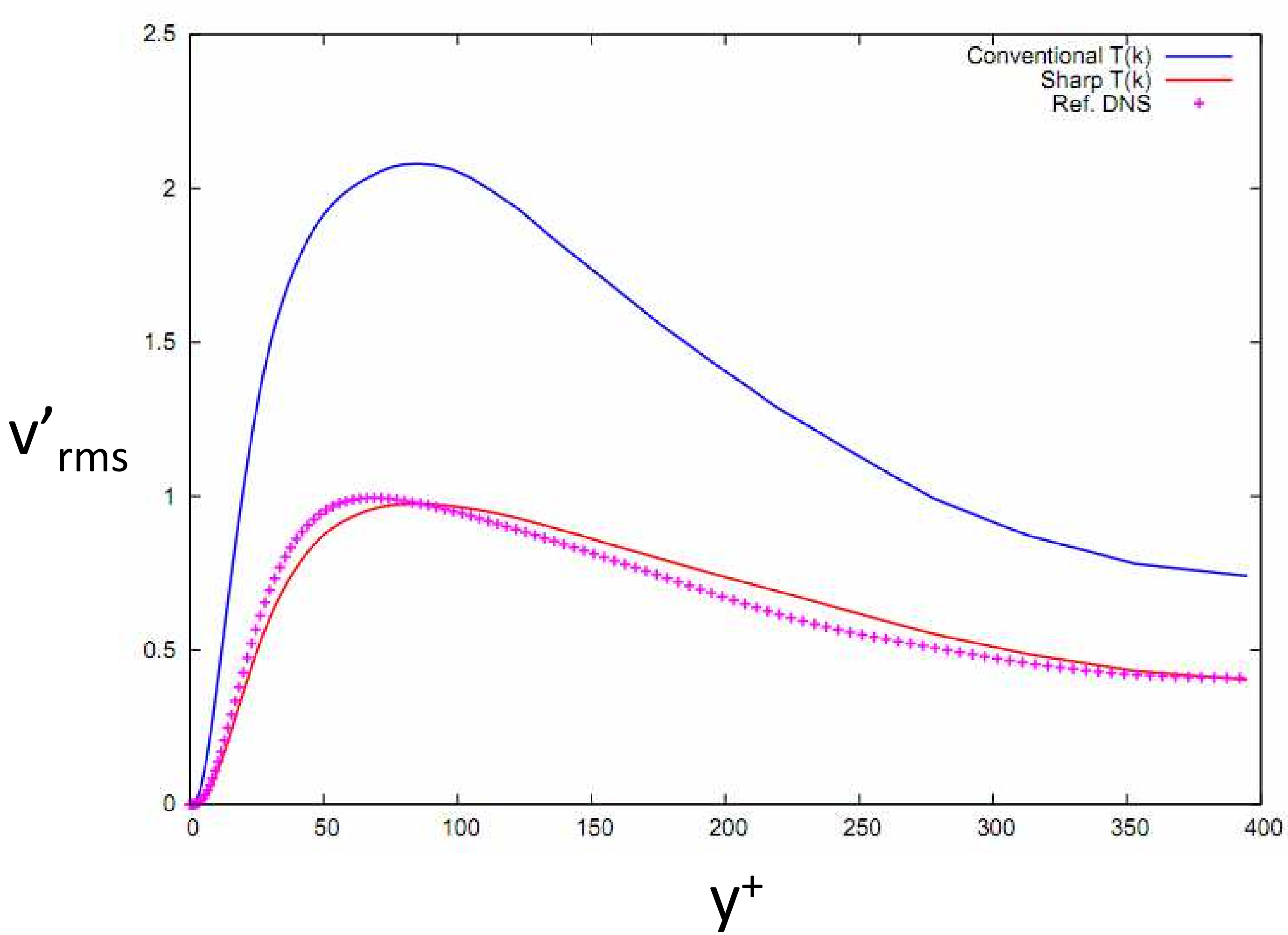
Symbols: 128x129x128 Fourier^2-Chebyshev KMM (1987)



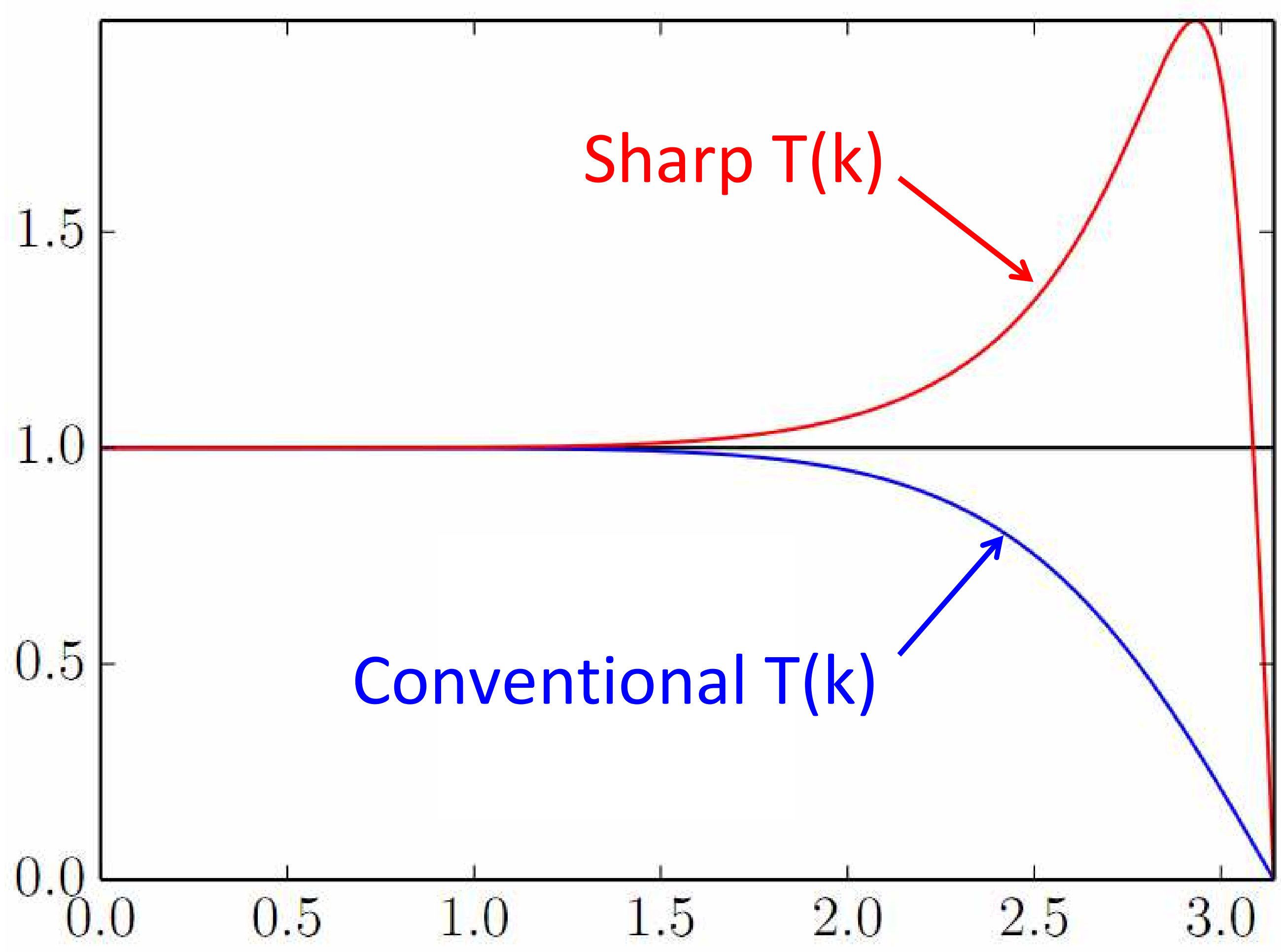
…but the use of sixth-order numerical dissipation is mandatory!



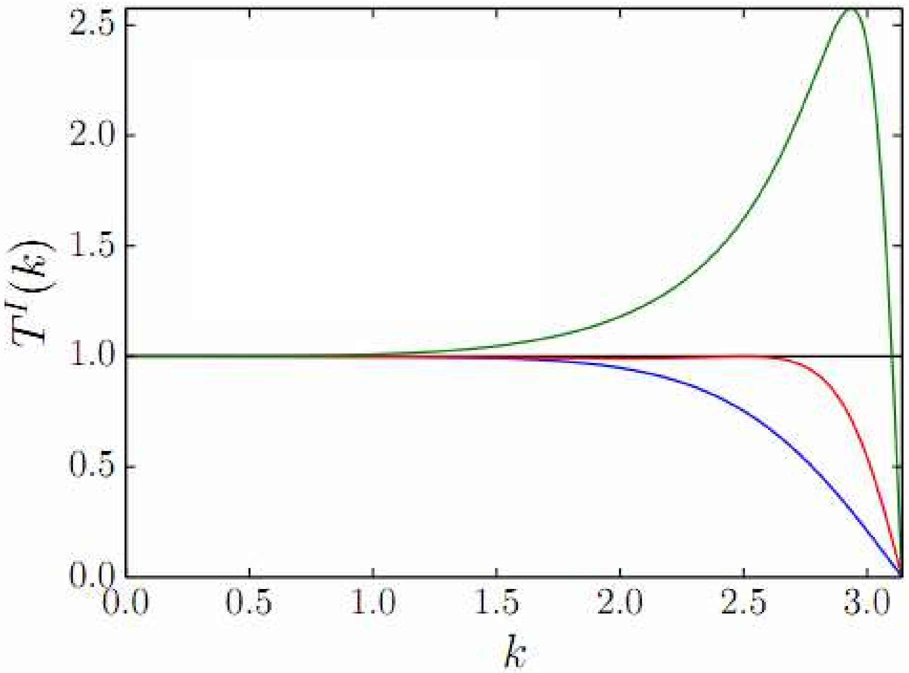
Example of LES at marginal resolution

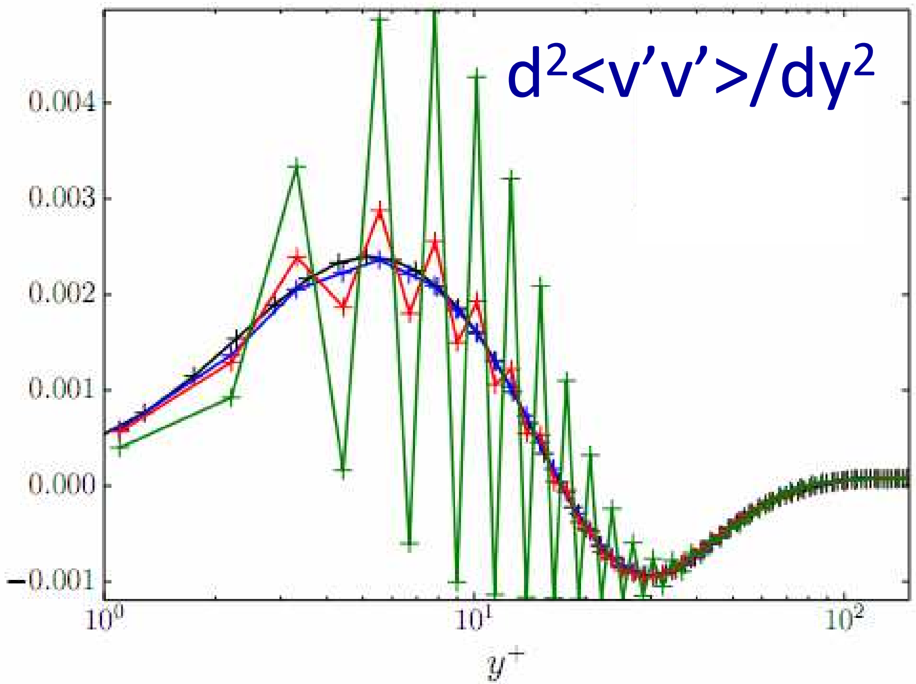


Conventional T(k) ↔ Sharp T(k)



Sharp T(k) successful, but…





…with the presence of very small amplitude oscillations on $<v'v'>$ that can be detected on the viscous diffusion term $d^2<v'v'>/dy^2$ in its budget.

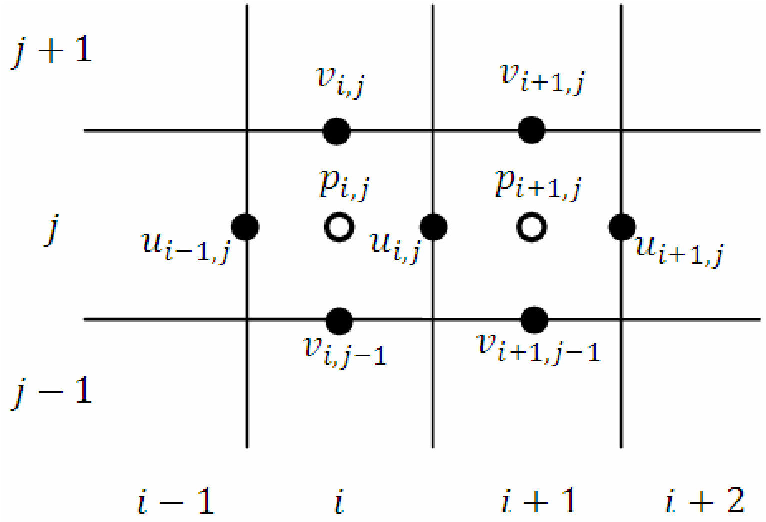
These oscillations are present even in DNS, and in a very attenuated form with conventional T(k).

Interpretation

- Quasi-singular modes, associated with the use of a staggered mesh only for the pressure, are the Achilles' heel of Incompact3d.
- They play against the robustness of the code when the spatial resolution is marginal.
- High-order numerical dissipation can restore stability and accuracy at marginal resolution but only for DNS.
- Sharp interpolation (highly sensitive at small scale) allows robust LES but with the presence spurious oscillations in the near-wall region.
- The sensitivity of results with respect to interpolation confirms that <u>the pressure treatment is the key point</u>.
- Two features of the pressure treatment can be suspected to explain the present difficulties:
  - the mesh organization (only partially staggered)
  - the boundary condition on pressure (homogenous Neumann type)

Move to fully staggered mesh

- Advantages
  - No quasi-singular modes
  - Same compact schemes
  - Poisson solver OK
- Drawbacks
  - Less simple and original
  - Development of boundary conditions for staggered FD schemes (only ncl=1 is available)
  - No clear statement about numerical stability
  - Could increase a bit the computational cost

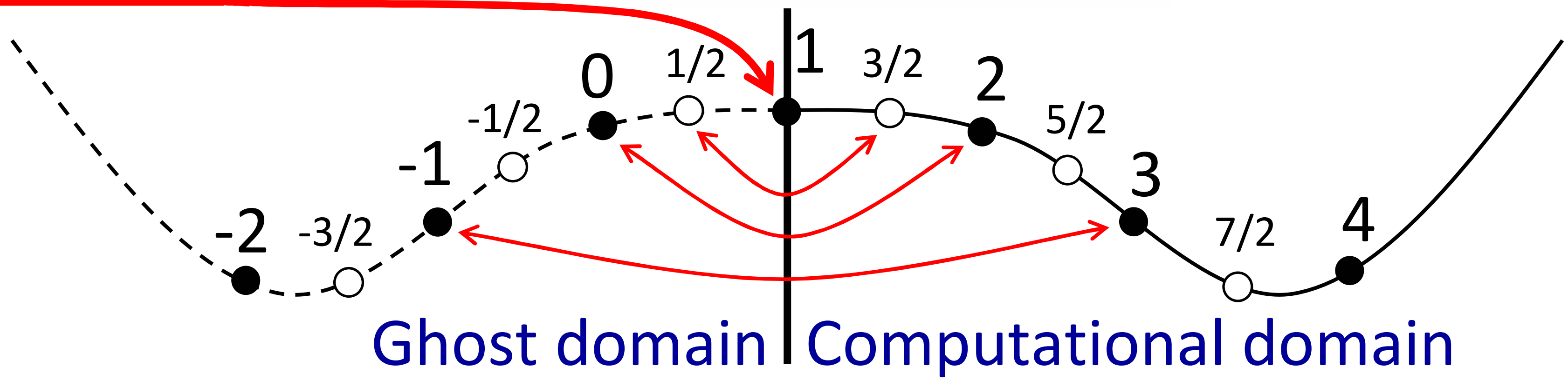


Pressure boundary condition

- The spectral Poisson solver requires to assume an homogeneous Neumann condition
  - ghost boundary condition used to ensure the equivalence between FD in physical and spectral spaces.
  - classical assumption in the context of the projection method (incompressibility condition).
  - second-order accurate in space.
  - the homogeneous Neumann condition is included in the (staggered) first derivative compact FD scheme.
  - the divergence operator is defined consistently through a compatible ghost boundary condition.

Ghost boundary conditions (symmetry)

$f'(0)=0$ and $f_0 \rightarrow f_2,\ f_{-1} \rightarrow f_3,\ f'_{-1/2} \rightarrow -f'_{3/2}$



Conventional staggered FD ☺

$$f'_{i+1/2} = \frac{f_{i+1} - f_i}{\Delta x}$$

No ghost boundary required → O(Δx^2)

Compact staggered FD 😐

$$\alpha f'_{i-1/2} + f'_{i+1/2} + \alpha f'_{i+3/2} = a\frac{f_{i+1} - f_i}{\Delta x} + b\frac{f_{i+2} - f_{i-1}}{3\Delta x}$$

Ghost boundary required → **O(Δx^2)+~~Gibbs~~!**

Time advancement

Explicit time integration (AB,RK) and fractional step method

$$\frac{\mathbf{u}^* - \mathbf{u}^k}{\Delta t} = a_k \mathbf{F}^k + b_k \mathbf{F}^{k-1} - c_k \boldsymbol{\nabla} \tilde{p}^k$$

$$\frac{\mathbf{u}^{**} - \mathbf{u}^*}{\Delta t} = c_k \boldsymbol{\nabla} \tilde{p}^k$$

$$\frac{\mathbf{u}^{k+1} - \mathbf{u}^{**}}{\Delta t} = -c_k \boldsymbol{\nabla} \tilde{p}^{k+1}$$

with

$$\mathbf{F}^k = -\frac{1}{2}[\nabla(\mathbf{u}^k \otimes \mathbf{u}^k) + (\mathbf{u}^k.\nabla)\mathbf{u}^k] + \nu \nabla^2 \mathbf{u}^k$$

$$\tilde{p}^{k+1} = \frac{1}{c_k \Delta t} \int_{t_k}^{t_{k+1}} p \; dt, \quad \tilde{\mathbf{f}}^{k+1} = \frac{1}{c_k \Delta t} \int_{t_k}^{t_{k+1}} \mathbf{f} \; dt$$

Pressure boundary condition

- The incompressibility condition

$$\nabla.\mathbf{u}^{k+1} = 0$$

  leads to the Poisson equation

$$\nabla.\nabla\tilde{p}^{k+1} = \frac{\nabla.\mathbf{u}^{**}}{c_k \Delta t}$$

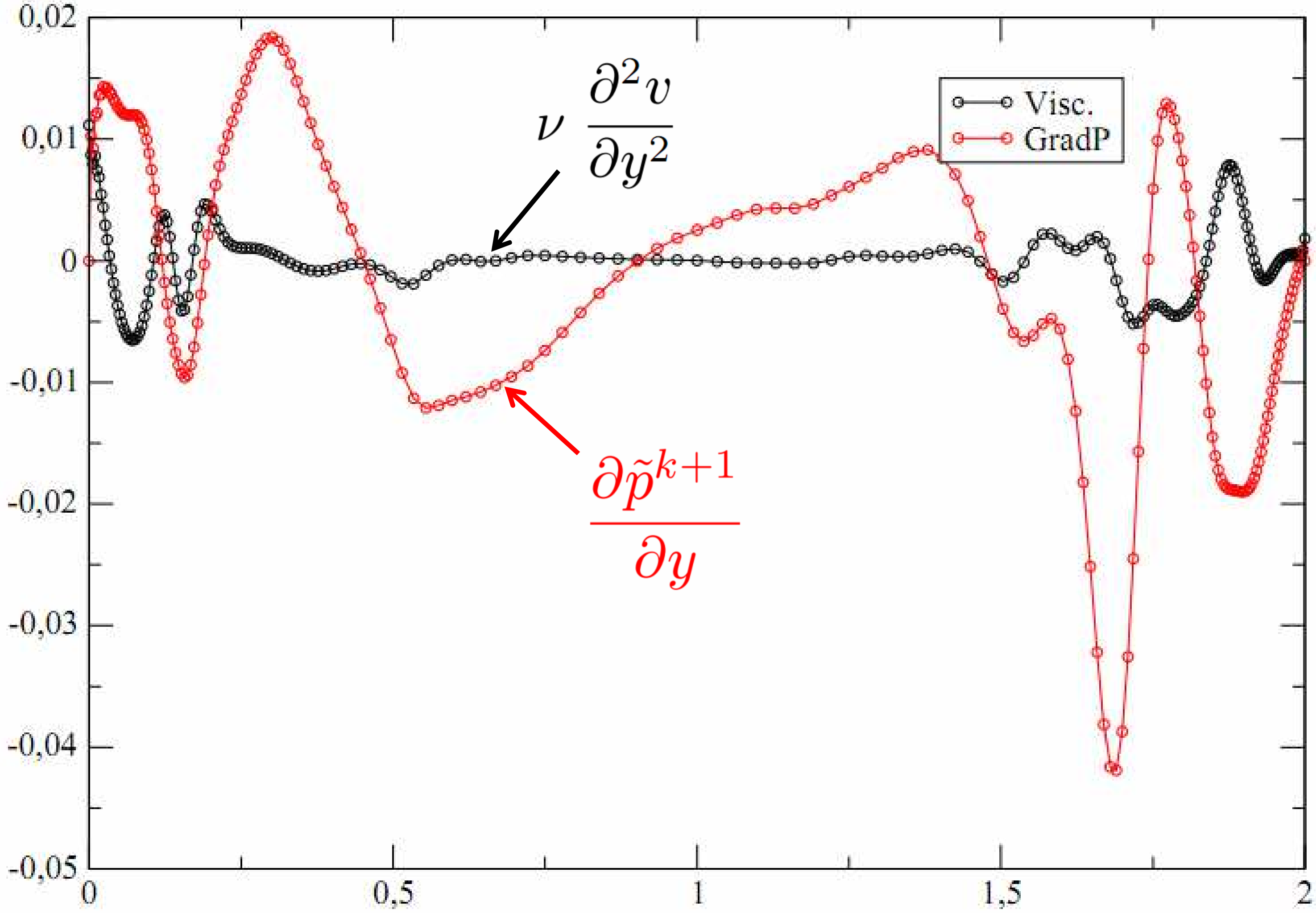
  where the pressure is assumed to check homogeneous boundary conditions.

- For instance, if a no-slip boundary condition is imposed at $y=\pm L_y/2$, we assume

$$\underbrace{\left.\frac{\partial \tilde{p}^{k+1}}{\partial y}\right|_{\pm L_y/2} = 0}_{\text{approx. pressure BC}} \leftrightarrow \underbrace{\left.\frac{\partial p}{\partial y}\right|_{\pm L_y/2} = \nu \left.\frac{\partial^2 v}{\partial y^2}\right|_{\pm L_y/2}}_{\text{consistent pressure BC}}$$

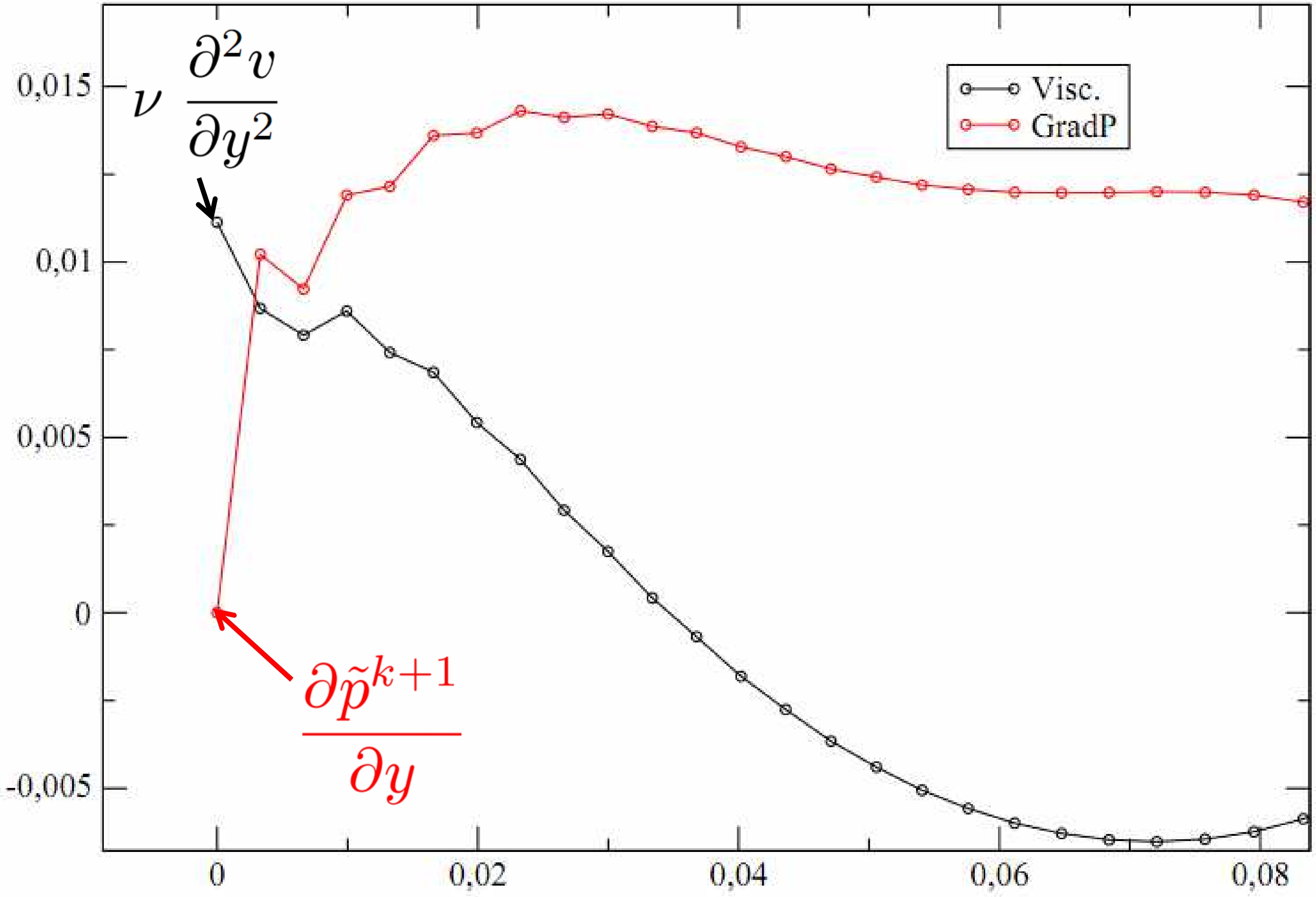
Pressure boundary condition

Instantaneous profiles in a turbulent channel



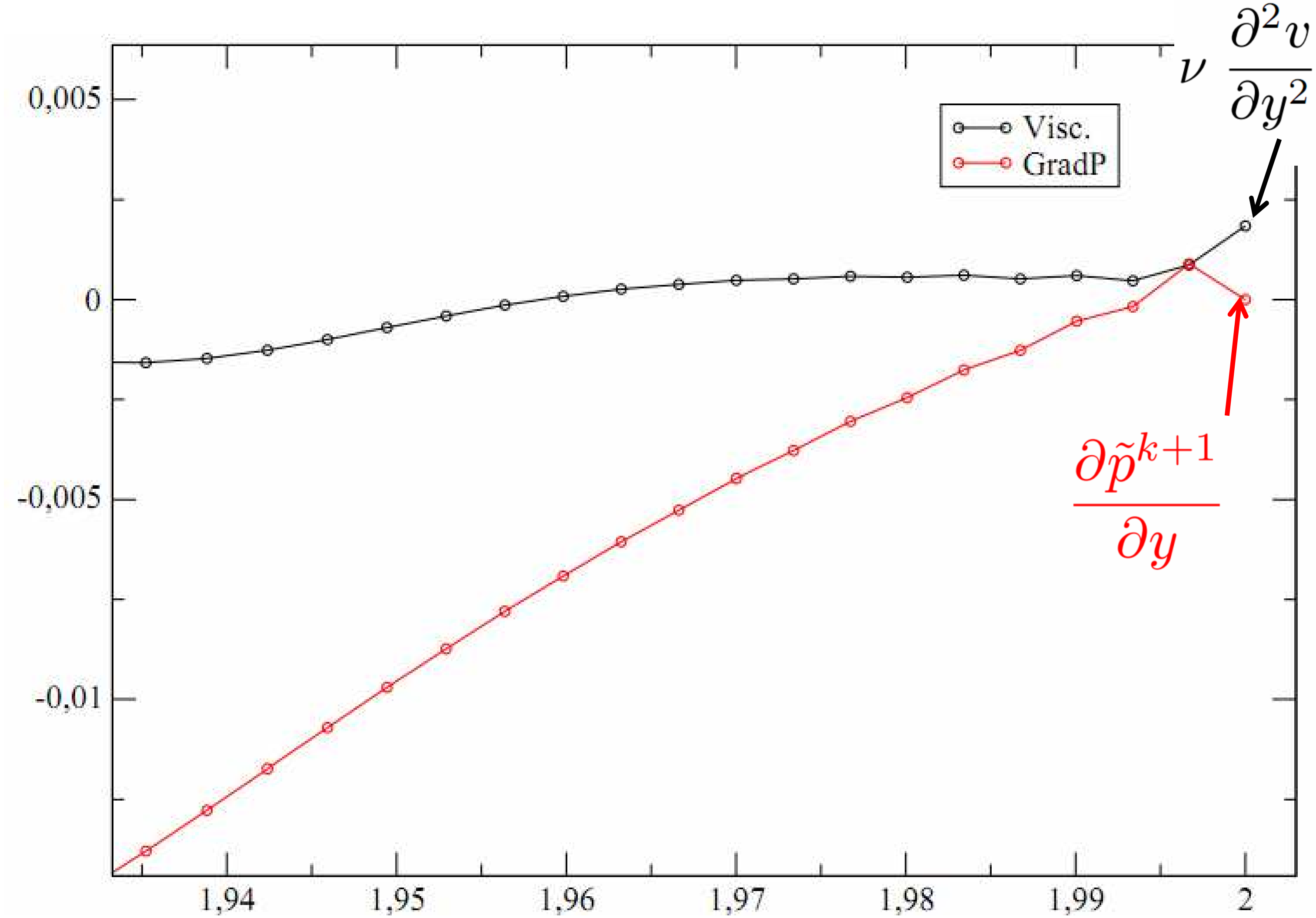
Pressure boundary condition

Instantaneous profiles in a turbulent channel



Pressure boundary condition

Instantaneous profiles in a turbulent channel



Improvement of the pressure BC

- Adapt the "incremental pressure-correction scheme"

$$\frac{1}{2\Delta t}(3\tilde{u}^{k+1} - 4u^k + u^{k-1}) - \nu\nabla^2\tilde{u}^{k+1} + \nabla p^k = f(t^{k+1}), \quad \tilde{u}^{k+1}|_\Gamma = 0,$$

$$\begin{cases} \frac{1}{2\Delta t}(3u^{k+1} - 3\tilde{u}^{k+1}) + \nabla\phi^{k+1} = 0, \\ \nabla \cdot u^{k+1} = 0, \quad u^{k+1} \cdot n|_\Gamma = 0, \end{cases}$$

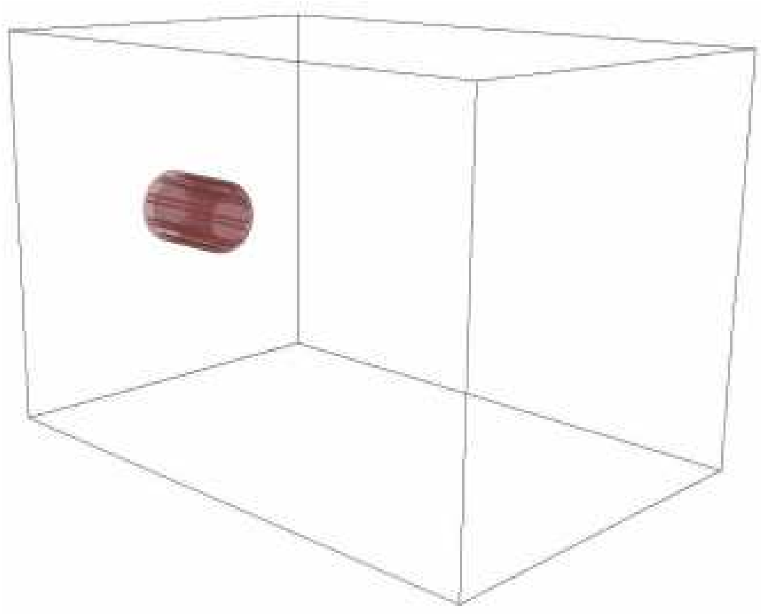
$$\phi^{k+1} = p^{k+1} - p^k + \nu\nabla \cdot \tilde{u}^{k+1}.$$

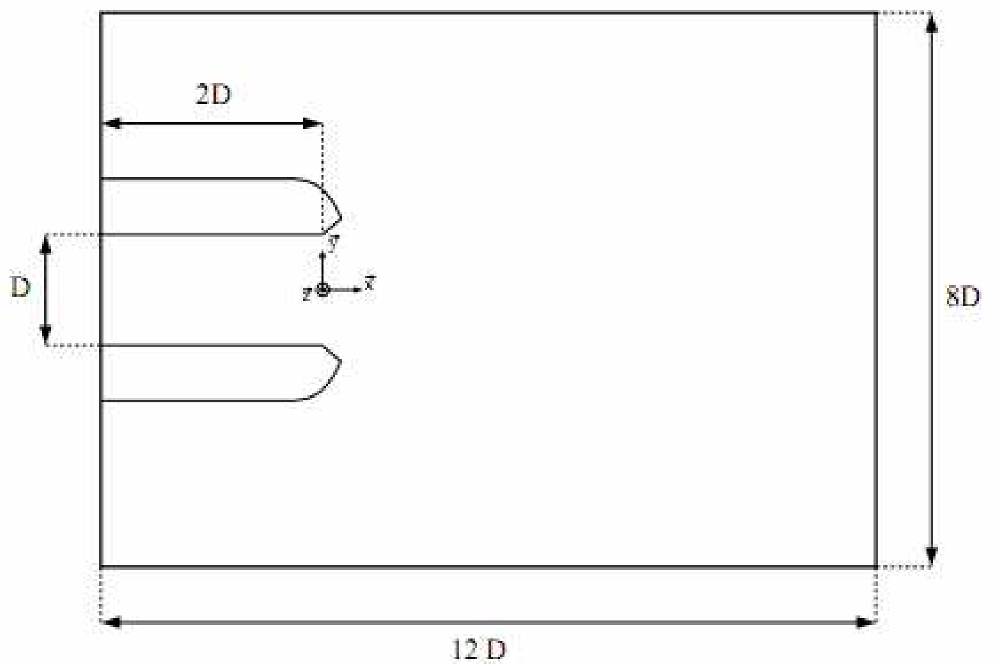
→ consistent BC $\boxed{\partial_n p^{k+1}|_\Gamma = (f(t^{k+1}) - \nu\nabla \times \nabla \times u^{k+1}) \cdot n|_\Gamma,}$

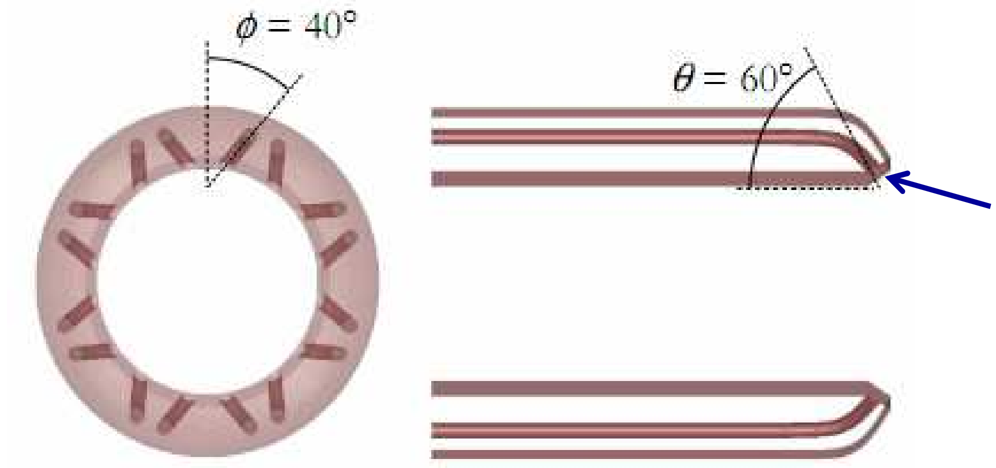
to an <u>explicit or semi-explicit</u> time advancement.

- Find a time advancement based on two (or more) prediction steps and one final projection (in progress).

Mass conservation IBM







Microchannel diameter ≈8Δx
→ loss of flow rate at marginal resolution
→ use of ε^{-}

Mass conservation IBM

Conventional time advancement

$$\frac{u_i^* - u_i^n}{\Delta t} = \frac{3}{2}F_i^n - \frac{1}{2}F_i^{n-1} - \frac{1}{\rho}\frac{\partial p^{n-\frac{1}{2}}}{\partial x_i}$$

$$\frac{u_i^{**} - u_i^*}{\Delta t} = \frac{1}{\rho}\frac{\partial p^{n-\frac{1}{2}}}{\partial x_i}$$

$$\frac{u_i^{n+1} - u_i^{**}}{\Delta t} = -\frac{1}{\rho}\frac{\partial p^{n+\frac{1}{2}}}{\partial x_i}$$

← Forcing on u_i^* where ε=1

Specific Poisson equation

$$\frac{1}{\rho}\frac{\partial^2 p^{n+\frac{1}{2}}}{\partial x_i \partial x_i} = \frac{1}{\Delta t}\frac{\partial\left((1-\varepsilon^-)\ {}^{*}_{i}\right)}{\partial x_i}$$

← ε^- “retracted” by a mesh size

Only a problem at marginal resolution!

(see Lamballais, JFM, 2014 for channel with IBM)

Further challenges (1/3)

- 2D version of Incompact3d: either with MPI or with OpenMP (Porto Alegre, Brazil)
- Stretching in 2 directions: No direct Poisson solver but iterative method in spectral space (Buenos Aires, Argentina)
- Free surface (Porto Alegre, Brazil)

Further challenges (2/3)

- Quasicompact3d and Compact3d: dCSE project with NAG for implementation of 2D decomp & FFT in Compact3d (NAG and Poitiers)
- Multiblock domain strategy (within UKTC, Charles Moulinec, Daresbury)
- Hybrid approach: best strategy to be discuss depending on hardware

Further challenges (3/3)

- How to deal with user requests? (10 to 15 emails every month)
- How to validate and integrate new developments in main version of the code? (benchmarks procedure)
- How to keep the website up to date? (1 or 2 releases every year)
- Any questions?