Calculating, modelling and understanding turbulence using adaptive wavelet simulation

Nicholas Kevlahan

Department of Mathematics & Statistics

McMaster University

December 2, 2005

Collaborators

- Oleg Vasilyev
  University of Colorado at Boulder
- Dan Goldstein
  University of Colorado at Boulder
- Jahrul Alam
  McMaster University (PhD student)

Outline

What is turbulence?

Smooth flow is not possible for $Re = UL/\nu \gg 1$, and at $Re = O(10^4)$ it becomes turbulent:

- Flow is characterized by intense localized three-dimensional vorticity.
- Flow is unsteady, and its energy is intermittent in space and scale.
- Fluid properties (diffusion, momentum transfer, drag, lift etc.) differ dramatically from those of laminar flow.

What is turbulence?

Smooth flow is not possible for $Re = UL/\nu \gg 1$, and at $Re = O(10^4)$ it becomes turbulent:

- Flow is characterized by intense localized three-dimensional vorticity.
- Flow is unsteady, and its energy is intermittent in space and scale.
- Fluid properties (diffusion, momentum transfer, drag, lift etc.) differ dramatically from those of laminar flow.

What is turbulence?

Smooth flow is not possible for $Re = UL/\nu \gg 1$, and at $Re = O(10^4)$ it becomes turbulent:

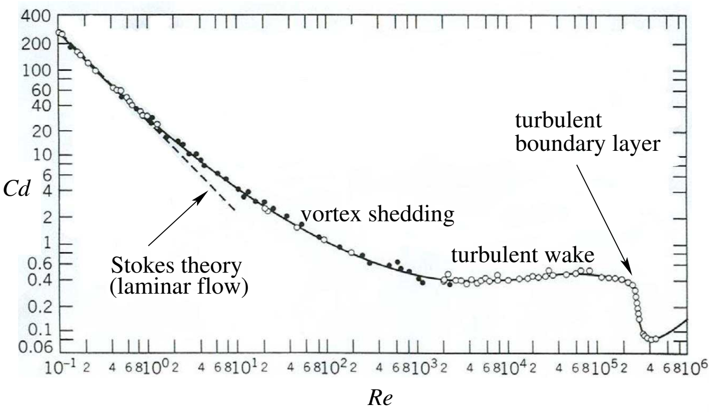
- Flow is characterized by intense localized three-dimensional vorticity.
- Flow is unsteady, and its energy is intermittent in space and scale.
- Fluid properties (diffusion, momentum transfer, drag, lift etc.) differ dramatically from those of laminar flow.

What is turbulence?

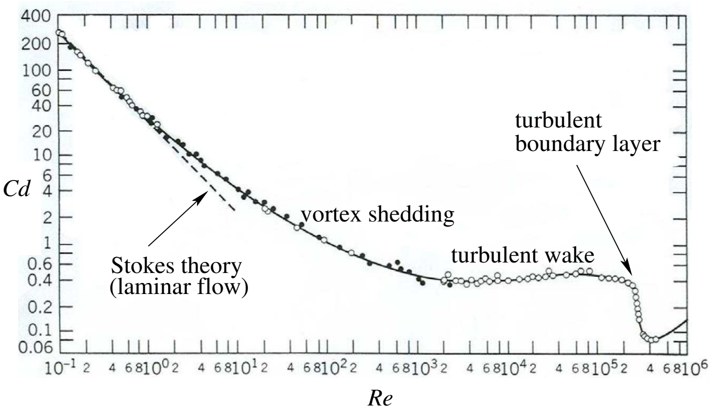
Smooth flow is not possible for $Re = UL/\nu \gg 1$, and at $Re = O(10^4)$ it becomes turbulent:

- Flow is characterized by intense localized three-dimensional vorticity.
- Flow is unsteady, and its energy is intermittent in space and scale.
- Fluid properties (diffusion, momentum transfer, drag, lift etc.) differ dramatically from those of laminar flow.

Drag on a sphere: effect of turbulence

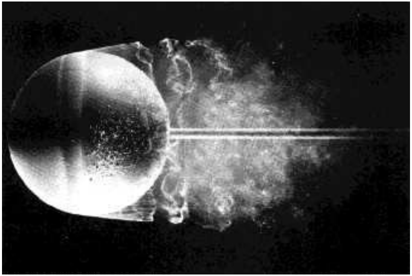


Drag on a sphere: effect of turbulence

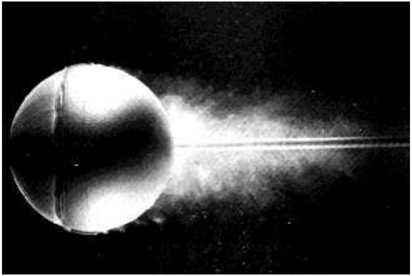


We should be able to calculate this curve!

Drag on a sphere: effect of turbulence



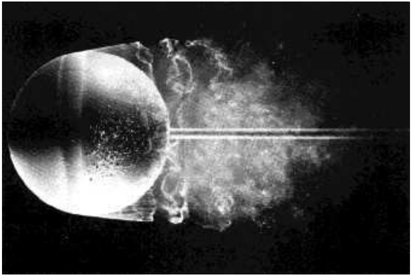
$Re = 15\,000$
turbulent wake



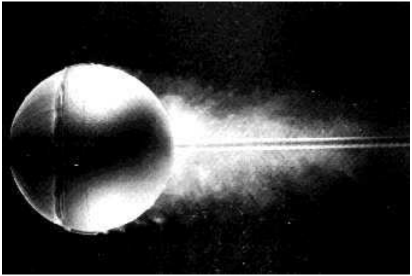
$Re = 30\,000$
turbulent boundary layer (tripped)

(ONERA, Werle 1980 from www.efluids.net)

Drag on a sphere: effect of turbulence



$Re = 15\,000$
turbulent wake



$Re = 30\,000$
turbulent boundary layer (tripped)

(ONERA, Werle 1980 from www.efluids.net)

We would also like to control turbulence.

The challenges of turbulence

1. No general theory of turbulence exists.
2. Few theorems have been proved for the Navier–Stokes equations on bounded domains at high Reynolds numbers.
3. Model-free computations are limited to moderate Reynolds numbers ($O(10^4)$) and simple geometries (homogeneous isotropic turbulence or channel flows).
4. Experiments are expensive, and produce only a partial picture of the flow.

The challenges of turbulence

1. No general theory of turbulence exists.
2. Few theorems have been proved for the Navier–Stokes equations on bounded domains at high Reynolds numbers.
3. Model-free computations are limited to moderate Reynolds numbers ($O(10^4)$) and simple geometries (homogeneous isotropic turbulence or channel flows).
4. Experiments are expensive, and produce only a partial picture of the flow.

The challenges of turbulence

1. No general theory of turbulence exists.
2. Few theorems have been proved for the Navier–Stokes equations on bounded domains at high Reynolds numbers.
3. Model-free computations are limited to moderate Reynolds numbers ($O(10^4)$) and simple geometries (homogeneous isotropic turbulence or channel flows).
4. Experiments are expensive, and produce only a partial picture of the flow.

The challenges of turbulence

1. No general theory of turbulence exists.
2. Few theorems have been proved for the Navier–Stokes equations on bounded domains at high Reynolds numbers.
3. Model-free computations are limited to moderate Reynolds numbers ($O(10^4)$) and simple geometries (homogeneous isotropic turbulence or channel flows).
4. Experiments are expensive, and produce only a partial picture of the flow.

The challenges of turbulence

1. No general theory of turbulence exists.
2. Few theorems have been proved for the Navier–Stokes equations on bounded domains at high Reynolds numbers.
3. Model-free computations are limited to moderate Reynolds numbers ($O(10^4)$) and simple geometries (homogeneous isotropic turbulence or channel flows).
4. Experiments are expensive, and produce only a partial picture of the flow.

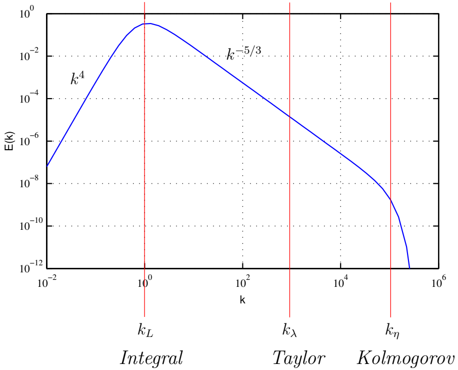
The challenges of turbulence

1. No general theory of turbulence exists.
2. Few theorems have been proved for the Navier–Stokes equations on bounded domains at high Reynolds numbers.
3. Model-free computations are limited to moderate Reynolds numbers ($O(10^4)$) and simple geometries (homogeneous isotropic turbulence or channel flows).
4. Experiments are expensive, and produce only a partial picture of the flow.

Unfortunately, most engineering, geophysical and physiological flows are strongly turbulent!

Why is turbulence difficult to compute?

Turbulence energy spectrum at $Re \approx 10^6$



Continuous range of active wavenumbers that increases with Reynolds number: $k_\eta/k_L \sim Re^{3/4}$.

Upper bound on computational complexity

- Number of spatial degrees of freedom is $(L/\eta)^3 \sim Re^{9/4}$.
- If $\Delta t \sim \eta/L$, total computational complexity is $\propto Re^3$.
- For typical aerodynamical flows $Re \approx 10^6$, and we might expect $O(10^{18})$ computational elements!
- Is this the best we can hope for?

Upper bound on computational complexity

- Number of spatial degrees of freedom is $(L/\eta)^3 \sim Re^{9/4}$.
- If $\Delta t \sim \eta/L$, total computational complexity is $\propto Re^3$.
- For typical aerodynamical flows $Re \approx 10^6$, and we might expect $O(10^{18})$ computational elements!
- Is this the best we can hope for?

Upper bound on computational complexity

- Number of spatial degrees of freedom is $(L/\eta)^3 \sim Re^{9/4}$.
- If $\Delta t \sim \eta/L$, total computational complexity is $\propto Re^3$.
- For typical aerodynamical flows $Re \approx 10^6$, and we might expect $O(10^{18})$ computational elements!
- Is this the best we can hope for?

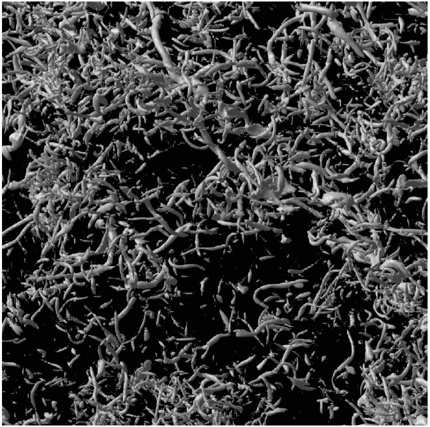
Upper bound on computational complexity

- Number of spatial degrees of freedom is $(L/\eta)^3 \sim Re^{9/4}$.
- If $\Delta t \sim \eta/L$, total computational complexity is $\propto Re^3$.
- For typical aerodynamical flows $Re \approx 10^6$, and we might expect $O(10^{18})$ computational elements!
- Is this the best we can hope for?

Upper bound on computational complexity

- Number of spatial degrees of freedom is $(L/\eta)^3 \sim Re^{9/4}$.
- If $\Delta t \sim \eta/L$, total computational complexity is $\propto Re^3$.
- For typical aerodynamical flows $Re \approx 10^6$, and we might expect $O(10^{18})$ computational elements!
- Is this the best we can hope for?

Turbulence is intermittent in space and time



DNS of homogeneous isotropic turbulence at $\mathrm{Re}_\lambda = 1\,217$ (Yokokawa et al. 2002). Vortices are intermittent and multiscale.

Hypotheses for efficient simulation of turbulence

1. Turbulence is highly intermittent in both space and time: the actual number of degrees of freedom is much less than $O(Re^3)$.
2. Turbulence may be divided into an organized part (coherent vortices), and a stochastic part (random noise).
3. The coherent vortices must be resolved, but the noise may be modelled (or neglected entirely).
4. The coherent vortices may be extracted using adaptive wavelet filtering.

Hypotheses for efficient simulation of turbulence

1. Turbulence is highly intermittent in both space and time: the actual number of degrees of freedom is much less than $O(Re^3)$.
2. Turbulence may be divided into an organized part (coherent vortices), and a stochastic part (random noise).
3. The coherent vortices must be resolved, but the noise may be modelled (or neglected entirely).
4. The coherent vortices may be extracted using adaptive wavelet filtering.

Hypotheses for efficient simulation of turbulence

1. Turbulence is highly intermittent in both space and time: the actual number of degrees of freedom is much less than $O(Re^3)$.
2. Turbulence may be divided into an organized part (coherent vortices), and a stochastic part (random noise).
3. The coherent vortices must be resolved, but the noise may be modelled (or neglected entirely).
4. The coherent vortices may be extracted using adaptive wavelet filtering.

Hypotheses for efficient simulation of turbulence

1. Turbulence is highly intermittent in both space and time: the actual number of degrees of freedom is much less than $O(Re^3)$.
2. Turbulence may be divided into an organized part (coherent vortices), and a stochastic part (random noise).
3. The coherent vortices must be resolved, but the noise may be modelled (or neglected entirely).
4. The coherent vortices may be extracted using adaptive wavelet filtering.

Hypotheses for efficient simulation of turbulence

1. Turbulence is highly intermittent in both space and time: the actual number of degrees of freedom is much less than $O(Re^3)$.
2. Turbulence may be divided into an organized part (coherent vortices), and a stochastic part (random noise).
3. The coherent vortices must be resolved, but the noise may be modelled (or neglected entirely).
4. The coherent vortices may be extracted using adaptive wavelet filtering.

Hypotheses for efficient simulation of turbulence

1. Turbulence is highly intermittent in both space and time: the actual number of degrees of freedom is much less than $O(Re^3)$.
2. Turbulence may be divided into an organized part (coherent vortices), and a stochastic part (random noise).
3. The coherent vortices must be resolved, but the noise may be modelled (or neglected entirely).
4. The coherent vortices may be extracted using adaptive wavelet filtering.

This is the basis of Coherent Vortex Simulation which was developed in collaboration with Marie Farge and Kai Schneider.

Three levels of adaptivity:

1. Adaptive time step.
2. Dynamically adaptive grid.
3. Simultaneous space–time adaptivity.

Three levels of adaptivity:

1. Adaptive time step.
2. Dynamically adaptive grid.
3. Simultaneous space–time adaptivity.

Three levels of adaptivity:

1. Adaptive time step.
2. Dynamically adaptive grid.
3. Simultaneous space–time adaptivity.

Three levels of adaptivity:

1. Adaptive time step.
2. Dynamically adaptive grid.
3. Simultaneous space–time adaptivity.

Why use a wavelet basis for adaptivity?

- High rate of data compression (e.g. jpeg2 2000 image compression).
- Fast $O(\mathcal{N})$ transform.
- Fast signal de-noising (optimal for additive Gaussian noise).
- Easy to control wavelet properties (e.g. smoothness, boundary conditions).

Why use a wavelet basis for adaptivity?

- High rate of data compression (e.g. jpeg2 2000 image compression).
- Fast $O(\mathcal{N})$ transform.
- Fast signal de-noising (optimal for additive Gaussian noise).
- Easy to control wavelet properties (e.g. smoothness, boundary conditions).

Why use a wavelet basis for adaptivity?

- High rate of data compression (e.g. jpeg2 2000 image compression).
- Fast $O(\mathcal{N})$ transform.
- Fast signal de-noising (optimal for additive Gaussian noise).
- Easy to control wavelet properties (e.g. smoothness, boundary conditions).

Why use a wavelet basis for adaptivity?

- High rate of data compression (e.g. jpeg2 2000 image compression).
- Fast $O(\mathcal{N})$ transform.
- Fast signal de-noising (optimal for additive Gaussian noise).
- Easy to control wavelet properties (e.g. smoothness, boundary conditions).

Why use a wavelet basis for adaptivity?

- High rate of data compression (e.g. jpeg2 2000 image compression).
- Fast $O(\mathcal{N})$ transform.
- Fast signal de-noising (optimal for additive Gaussian noise).
- Easy to control wavelet properties (e.g. smoothness, boundary conditions).

Wavelet multiresolution analysis of $\mathbf{L}^2(\mathbb{R})$

A sequence of approximation subspaces
$\mathbf{M} = \{V^j \subset \mathbf{L}^2(\mathbb{R}) \mid j \in \mathcal{J}\}$ s.t.

- $V^j \subset V^{j+1}$ (subspaces are nested).
- $\cup_{j\in\mathcal{J}} V^j$ is dense in $\mathbf{L}^2(\mathbb{R})$.
- Each V^j has a Riesz basis of scaling functions $\{\phi^j_k \mid k \in \mathcal{K}^j\}$.

Wavelet multiresolution analysis of $\mathbf{L}^2(\mathbb{R})$

A sequence of approximation subspaces
$\mathbf{M} = \{V^j \subset \mathbf{L}^2(\mathbb{R}) \mid j \in \mathcal{J}\}$ s.t.

- $V^j \subset V^{j+1}$ (subspaces are nested).
- $\cup_{j\in\mathcal{J}} V^j$ is dense in $\mathbf{L}^2(\mathbb{R})$.
- Each V^j has a Riesz basis of scaling functions $\{\phi_k^j \mid k \in \mathcal{K}^j\}$.

Wavelet multiresolution analysis of $\mathbf{L}^2(\mathbb{R})$

A sequence of approximation subspaces
$\mathbf{M} = \{V^j \subset \mathbf{L}^2(\mathbb{R}) \mid j \in \mathcal{J}\}$ s.t.

- $V^j \subset V^{j+1}$ (subspaces are nested).
- $\cup_{j\in\mathcal{J}} V^j$ is dense in $\mathbf{L}^2(\mathbb{R})$.
- Each V^j has a Riesz basis of scaling functions $\{\phi_k^j \mid k \in \mathcal{K}^j\}$.

Wavelet multiresolution analysis of $\mathbf{L}^2(\mathbb{R})$

A sequence of approximation subspaces
$\mathbf{M} = \{V^j \subset \mathbf{L}^2(\mathbb{R}) \mid j \in \mathcal{J}\}$ s.t.

- $V^j \subset V^{j+1}$ (subspaces are nested).
- $\cup_{j\in\mathcal{J}} V^j$ is dense in $\mathbf{L}^2(\mathbb{R})$.
- Each V^j has a Riesz basis of scaling functions $\{\phi_k^j \mid k \in \mathcal{K}^j\}$.

Wavelet multiresolution analysis of $\mathbf{L}^2(\mathbb{R})$

A sequence of approximation subspaces
$\mathbf{M} = \{V^j \subset \mathbf{L}^2(\mathbb{R}) \mid j \in \mathcal{J}\}$ s.t.

- $V^j \subset V^{j+1}$ (subspaces are nested).
- $\cup_{j \in \mathcal{J}} V^j$ is dense in $\mathbf{L}^2(\mathbb{R})$.
- Each V^j has a Riesz basis of scaling functions $\{\phi_k^j \mid k \in \mathcal{K}^j\}$.

Wavelets ψ_k^j span the complement space W^j, where
$V^{j+1} = V^j \oplus W^j$, i.e. wavelet coefficients give the detail.

Biorthogonal second generation wavelets

- Constructed in the spatial domain.
- Form a collocation basis.
- Can be custom designed for complex domains and irregular sampling intervals.
- Wavelet transform is done in place.
- Both forward and inverse wavelet transforms exist on an adapted grid.
- Order of approximation can be varied easily.

Biorthogonal wavelets differ from orthogonal wavelets

- Not translations and dilations of a single wavelet.
- Form a Riesz basis of linearly independent vectors:

$$A||f||^2 \leq \sum_k |\langle f, \phi_k^j \rangle|^2 \leq B||f||^2,$$

where $A \leq 1 \leq B$. There is an associated dual basis $\{\tilde{\phi}_k^j\}$ s.t.

$$\frac{1}{B}||f||^2 \leq \sum_k |\langle f, \tilde{\phi}_k^j \rangle|^2 \leq \frac{1}{A}||f||^2,$$

and

$$\boxed{f = \sum_k \langle f, \tilde{\phi}_k^j \rangle \phi_k^j, \text{ where } \langle \phi_p^j, \tilde{\phi}_n^j \rangle = \delta[p-n].}$$

- Parseval relation does not hold.

Biorthogonal wavelets differ from orthogonal wavelets

- Not translations and dilations of a single wavelet.
- Form a Riesz basis of linearly independent vectors:

$$A||f||^2 \leq \sum_k |\langle f, \phi_k^j \rangle|^2 \leq B||f||^2,$$

where $A \leq 1 \leq B$. There is an associated dual basis $\{\tilde{\phi}_k^j\}$ s.t.

$$\frac{1}{B}||f||^2 \leq \sum_k |\langle f, \tilde{\phi}_k^j \rangle|^2 \leq \frac{1}{A}||f||^2,$$

and

$$\boxed{f = \sum_k \langle f, \tilde{\phi}_k^j \rangle \phi_k^j, \text{ where } \langle \phi_p^j, \tilde{\phi}_n^j \rangle = \delta[p-n].}$$

- Parseval relation does not hold.

Biorthogonal wavelets differ from orthogonal wavelets

- Not translations and dilations of a single wavelet.
- Form a Riesz basis of linearly independent vectors:

$$A||f||^2 \leq \sum_k |\langle f, \phi_k^j\rangle|^2 \leq B||f||^2,$$

where $A \leq 1 \leq B$. There is an associated dual basis $\{\tilde{\phi}_k^j\}$ s.t.

$$\frac{1}{B}||f||^2 \leq \sum_k |\langle f, \tilde{\phi}_k^j\rangle|^2 \leq \frac{1}{A}||f||^2,$$

and

$$\boxed{f = \sum_k \langle f, \tilde{\phi}_k^j\rangle \phi_k^j, \text{ where } \langle \phi_p^j, \tilde{\phi}_n^j\rangle = \delta[p-n].}$$

- Parseval relation does not hold.

Biorthogonal wavelets differ from orthogonal wavelets

- Not translations and dilations of a single wavelet.
- Form a Riesz basis of linearly independent vectors:

$$A||f||^2 \leq \sum_k |\langle f, \phi^j_k \rangle|^2 \leq B||f||^2,$$

where $A \leq 1 \leq B$. There is an associated dual basis $\{\tilde{\phi}^j_k\}$ s.t.

$$\frac{1}{B}||f||^2 \leq \sum_k |\langle f, \tilde{\phi}^j_k \rangle|^2 \leq \frac{1}{A}||f||^2,$$

and

$$\boxed{f = \sum_k \langle f, \tilde{\phi}^j_k \rangle \phi^j_k, \text{ where } \langle \phi^j_p, \tilde{\phi}^j_n \rangle = \delta[p-n].}$$

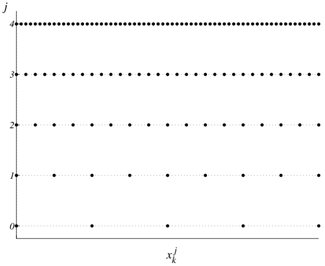
- Parseval relation does not hold.

Nested collocation wavelet grids

Scaling functions are constructed from interpolating polynomials of degree $2N-1$ on nested grids:

$$\mathcal{G}^j = \left\{ x_k^j \in \Omega : \; x_k^j = x_{2k}^{j+1}, \; k \in \mathcal{K}^j \right\}$$

Collocation: each scaling function and wavelet is associated to a unique grid point.

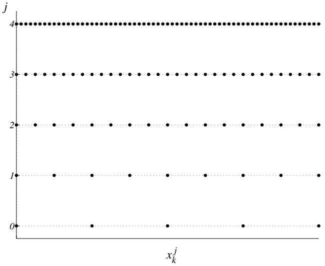


Nested collocation wavelet grids

Scaling functions are constructed from interpolating polynomials of degree $2N-1$ on nested grids:

$$\mathcal{G}^j = \left\{ x_k^j \in \Omega : \; x_k^j = x_{2k}^{j+1}, \; k \in \mathcal{K}^j \right\}$$

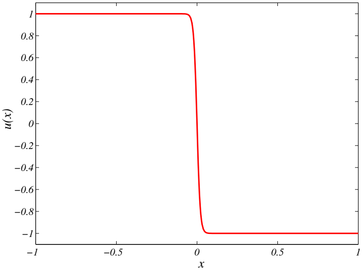
Collocation: each scaling function and wavelet is associated to a unique grid point.



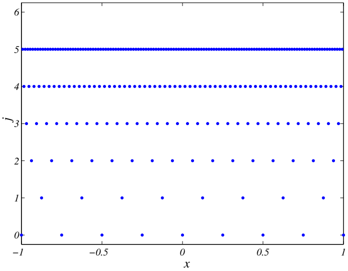
$$\boxed{u(x) = \sum_{k \in \mathcal{K}^J} u(x_k^J)\phi_k^J(x) = \sum_{k \in \mathcal{K}^0} u(x_k^0)\phi_k^0(x) + \sum_{j=0}^{J-1} \sum_{k \in \mathcal{L}^j} d_k^j \psi_k^j(x)}$$

Wavelet compression

$$u(x) = \sum_{k\in\mathcal{K}^0} u(x_k^0)\phi_k^0(x) + \sum_{j=0}^{+\infty}\sum_{k\in\mathcal{L}^j} d_k^j \psi_k^j(x)$$



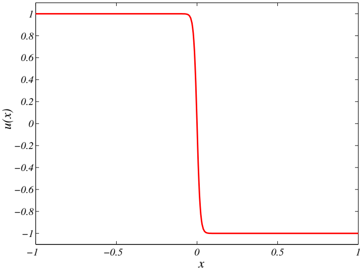
Function $u(x)$



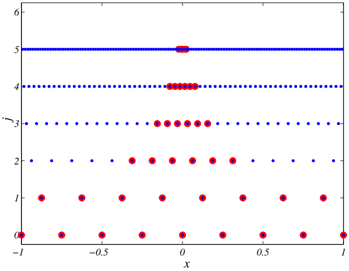
Wavelet locations x_k^j

Wavelet compression

$$u_{\geq}(x) = \sum_{k \in \mathcal{K}^0} u(x_k^0)\phi_k^0(x) + \sum_{j=0}^{J-1} \sum_{\substack{k \in \mathcal{L}^j \\ |d^j{}_k| \geq \epsilon}} d_k{}^j \psi_k{}^j(x)$$



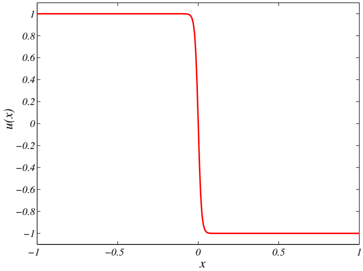
Function $u(x)$



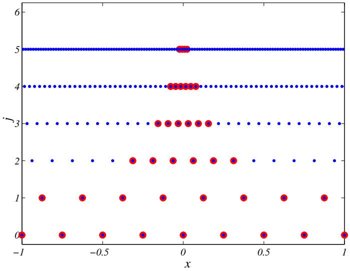
Wavelet locations $x_k{}^j$ $\epsilon = 10^{-3}$

Wavelet compression

$$\begin{aligned} ||u(x) - u_{\geq}(x)||_2 &= O(\epsilon) \\ \mathcal{N} &= O(\epsilon^{-1/2N}) \\ ||u(x) - u_{\geq}(x)||_2 &= O(\mathcal{N}^{-2N}) \end{aligned}$$



Function $u(x)$



Wavelet locations $x_k{}^j$ $\epsilon = 10^{-3}$

Solving a PDE by time marching

$$F\left(\frac{\partial u}{\partial t}, \frac{\partial^n u}{\partial x^n}, x, t\right) = 0$$

Solving a PDE by time marching

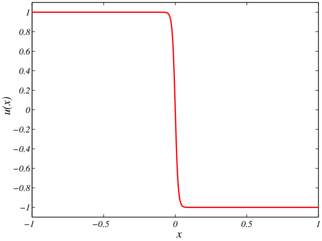
$$F\left(\frac{\partial u}{\partial t}, \frac{\partial^n u}{\partial x^n}, x, t\right) = 0$$

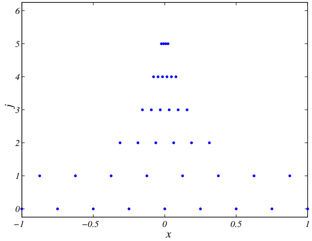
$$u(x^j{}_k) \Longrightarrow d_k{}^j \Longrightarrow \frac{\partial^n u}{\partial x^n}(x^j{}_k), \quad O(\mathcal{N}) \text{ complexity}$$

Solving a PDE by time marching

$$F\left(\frac{\partial u}{\partial t}, \frac{\partial^n u}{\partial x^n}, x, t\right) = 0$$

$$u(x^j{}_k) \Longrightarrow d_k{}^j \Longrightarrow \frac{\partial^n u}{\partial x^n}(x^j{}_k), \quad O(\mathcal{N}) \text{ complexity}$$

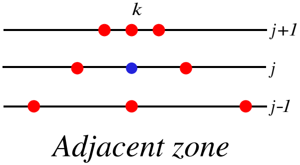


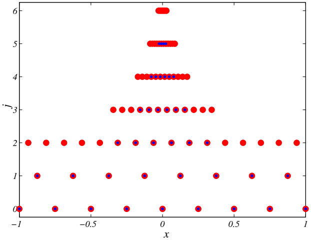


Solving a PDE by time marching

$$F\left(\frac{\partial u}{\partial t}, \frac{\partial^n u}{\partial x^n}, x, t\right) = 0$$

$$u(x^j{}_k) \Longrightarrow d_k{}^j \Longrightarrow \frac{\partial^n u}{\partial x^n}(x^j{}_k), \quad O(\mathcal{N}) \text{ complexity}$$

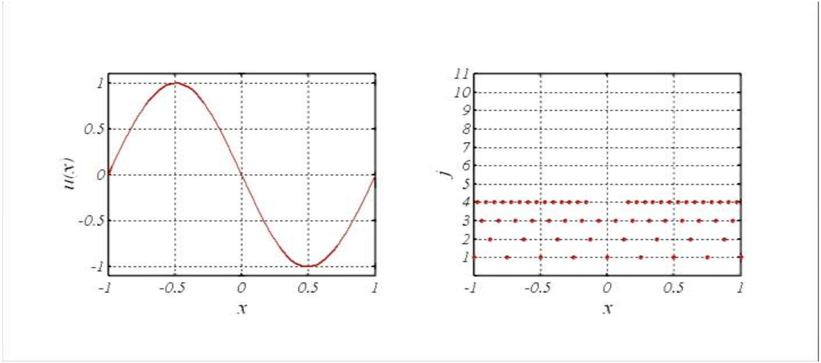




Burgers equation: steepening shock

$$\frac{\partial u}{\partial t} + u\frac{\partial u}{\partial x} = \nu\frac{\partial^2 u}{\partial x^2}, \quad x \in (-1, 1),\ t > 0,$$
$$u(x, 0) = -\sin(\pi x), \quad u(\pm 1, t) = 0$$

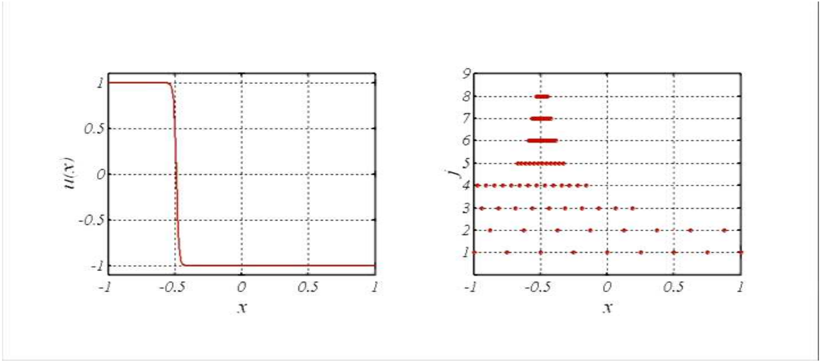
Parameters: $\nu = 10^{-2}/\pi$, $\epsilon = 10^{-5}$.



Burgers equation: moving shock

$$\frac{\partial u}{\partial t} + (1+u)\frac{\partial u}{\partial x} = \nu \frac{\partial^2 u}{\partial x^2}, \quad x \in (-\infty, \infty),\ t > 0,$$
$$u(x,0) = -\tanh((x+1/2)/(2\nu)), \quad u(\pm\infty, t) = \mp 1$$

Parameters: $\nu = 10^{-2}$, $\epsilon = 10^{-5}$.



Fluid–structure interaction

- Moderate to high Reynolds number flow around solid obstacles.
- Obstacle may be fixed, move or deform (e.g. in response to fluid forces).
- Applications: wind engineering of tall buildings, heat exchangers, underwater pipes, aeronautics.
- Example of spatial intermittency and complex geometry.

Fluid–structure interaction

- Moderate to high Reynolds number flow around solid obstacles.
- Obstacle may be fixed, move or deform (e.g. in response to fluid forces).
- Applications: wind engineering of tall buildings, heat exchangers, underwater pipes, aeronautics.
- Example of spatial intermittency and complex geometry.

Fluid–structure interaction

- Moderate to high Reynolds number flow around solid obstacles.
- Obstacle may be fixed, move or deform (e.g. in response to fluid forces).
- Applications: wind engineering of tall buildings, heat exchangers, underwater pipes, aeronautics.
- Example of spatial intermittency and complex geometry.

Fluid–structure interaction

- Moderate to high Reynolds number flow around solid obstacles.
- Obstacle may be fixed, move or deform (e.g. in response to fluid forces).
- Applications: wind engineering of tall buildings, heat exchangers, underwater pipes, aeronautics.
- Example of spatial intermittency and complex geometry.

Fluid–structure interaction

- Moderate to high Reynolds number flow around solid obstacles.
- Obstacle may be fixed, move or deform (e.g. in response to fluid forces).
- Applications: wind engineering of tall buildings, heat exchangers, underwater pipes, aeronautics.
- Example of spatial intermittency and complex geometry.

Fluid–structure interaction

Combine two methods:

1. Adaptive wavelet collocation for grid adaptation and derivatives.
2. Brinkman penalization to impose no-slip boundary conditions at the surface of an obstacle of arbitrary shape.

$$\frac{\partial \mathbf{u}}{\partial t} + (\mathbf{u} + \mathbf{U}) \cdot \nabla \mathbf{u} + \nabla P = \nu \Delta \mathbf{u} - \frac{1}{\eta}\chi(\mathbf{x}, t)(\mathbf{u} + \mathbf{U} - \mathbf{U}_o)$$

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

Fluid–structure interaction

Combine two methods:

1. Adaptive wavelet collocation for grid adaptation and derivatives.
2. Brinkman penalization to impose no-slip boundary conditions at the surface of an obstacle of arbitrary shape.

$$\frac{\partial \mathbf{u}}{\partial t} + (\mathbf{u} + \mathbf{U}) \cdot \nabla \mathbf{u} + \nabla P = \nu \Delta \mathbf{u} - \frac{1}{\eta} \chi(\mathbf{x}, t)(\mathbf{u} + \mathbf{U} - \mathbf{U}_o)$$

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

Fluid–structure interaction

Combine two methods:

1. Adaptive wavelet collocation for grid adaptation and derivatives.
2. Brinkman penalization to impose no-slip boundary conditions at the surface of an obstacle of arbitrary shape.

$$
\begin{aligned}
\frac{\partial \mathbf{u}}{\partial t} + (\mathbf{u} + \mathbf{U}) \cdot \nabla \mathbf{u} \quad + \quad \nabla P &= \nu \Delta \mathbf{u} \; - \frac{1}{\eta}\chi(\mathbf{x}, t)(\mathbf{u} + \mathbf{U} - \mathbf{U}_o) \\
\nabla \cdot \mathbf{u} \quad &= \quad 0
\end{aligned}
$$

Fluid–structure interaction

Combine two methods:

1. Adaptive wavelet collocation for grid adaptation and derivatives.
2. Brinkman penalization to impose no-slip boundary conditions at the surface of an obstacle of arbitrary shape.

$$\frac{\partial \mathbf{u}}{\partial t} + (\mathbf{u}+\mathbf{U})\cdot\nabla\mathbf{u} \quad + \quad \nabla P = \nu\Delta\mathbf{u} - \frac{1}{\eta}\chi(\mathbf{x},t)(\mathbf{u}+\mathbf{U}-\mathbf{U}_o)$$
$$\nabla\cdot\mathbf{u} \quad = \quad 0$$

Obstacle response is modelled as a damped harmonic oscillator

$$m\ddot{\mathbf{x}}_o(t) + b\dot{\mathbf{x}}_o(t) + k\mathbf{x}_o = \mathbf{F}(t).$$

Fluid–structure interaction: time scheme

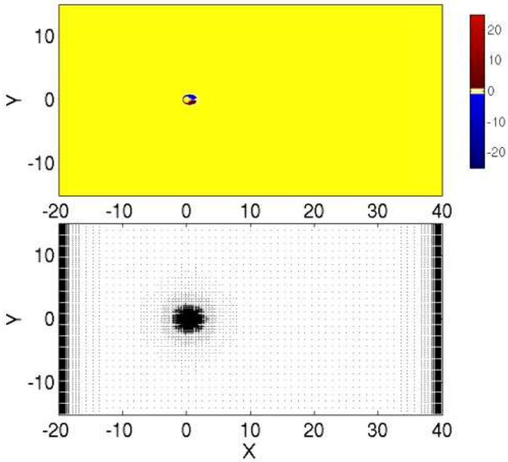
- Second order backwards difference.
- Semi-implicit discretization of convection term.
- Split-step to enforce divergence free velocity.

Fluid–structure interaction: time scheme

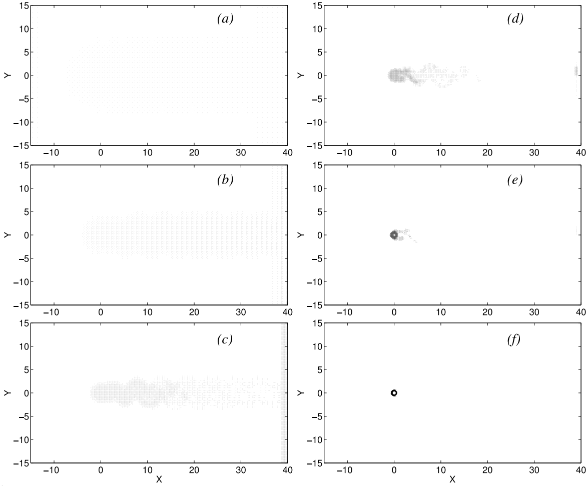
- Second order backwards difference.
- Semi-implicit discretization of convection term.
- Split-step to enforce divergence free velocity.

Poisson equation is solved using an adaptive wavelet multilevel method with V-cycles.

2D fluid–structure interaction: moving cylinder, $Re = 100$

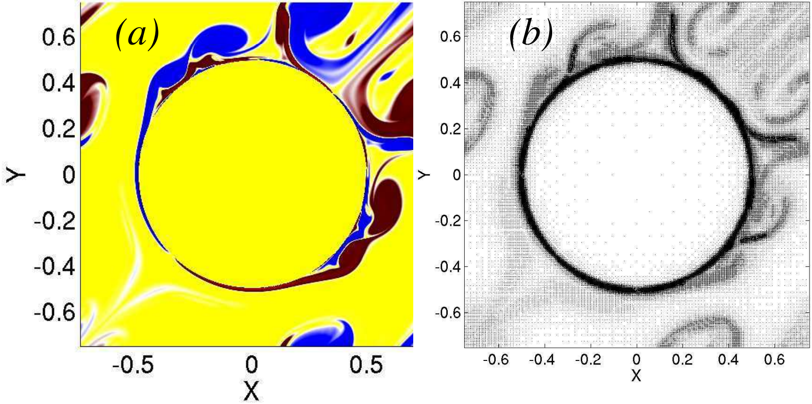


2D fluid–structure interaction: fixed cylinder, $Re = 100$



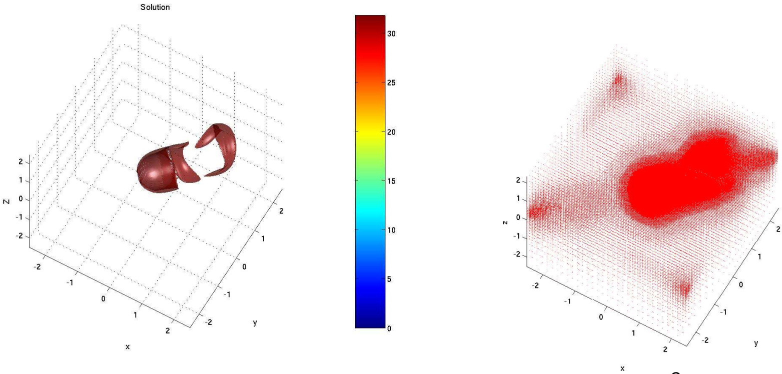
Grid at scales $j = 4$ to $j = 9$, compression ratio is $1/270$.

2D fluid–structure interaction



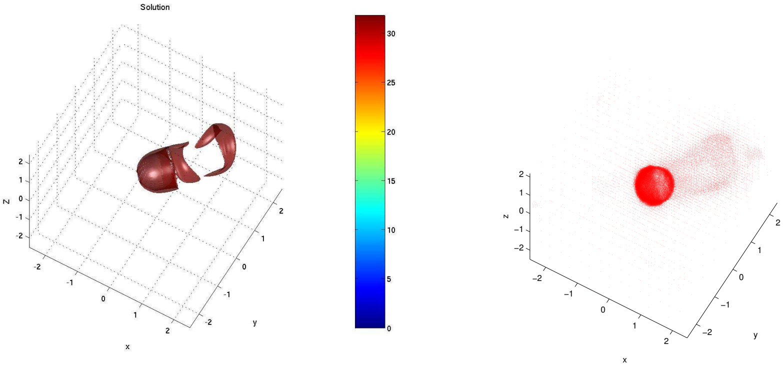
Periodic cylinder array at $Re = 10^4$, $t = 3.5$, 869^2. (a) Vorticity. (b) Grid.

3D fluid–structure interaction



Flow around a sphere at $Re = 550$, effective grid 256^3.
Vorticity isosurface (30% $||\omega||_\infty$) and grid at $t = 16$.

3D fluid–structure interaction



Flow around a sphere at $Re = 550$, effective grid 256^3.
Vorticity isosurface (30% $||\omega||_\infty$) and grid at $t = 16$.

Space–time adaptive wavelet PDE solver (with J. Alam)

Advantages

Open questions

Space–time adaptive wavelet PDE solver (with J. Alam)

Advantages

- Global error control in time.
- Local time step.
- Potentially optimal complexity for highly intermittent problems
- Grid reveals dynamics of problem.

Open questions

Space–time adaptive wavelet PDE solver (with J. Alam)

Advantages

- Global error control in time.
  Error grows uncontrollably in classical time marching.
- Local time step.
- Potentially optimal complexity for highly intermittent problems
- Grid reveals dynamics of problem.

Open questions

Space–time adaptive wavelet PDE solver (with J. Alam)

Advantages

- Global error control in time.
  Error grows uncontrollably in classical time marching.
- Local time step.
- Potentially optimal complexity for highly intermittent problems
- Grid reveals dynamics of problem.

Open questions

Space–time adaptive wavelet PDE solver (with J. Alam)

Advantages

- Global error control in time.
  Error grows uncontrollably in classical time marching.
- Local time step.
- Potentially optimal complexity for highly intermittent problems
- Grid reveals dynamics of problem.

Open questions

Space–time adaptive wavelet PDE solver (with J. Alam)

Advantages

- Global error control in time.
  Error grows uncontrollably in classical time marching.
- Local time step.
- Potentially optimal complexity for highly intermittent problems
- Grid reveals dynamics of problem.

Open questions

Space–time adaptive wavelet PDE solver (with J. Alam)

Advantages

- Global error control in time.
  Error grows uncontrollably in classical time marching.
- Local time step.
- Potentially optimal complexity for highly intermittent problems
- Grid reveals dynamics of problem.

Open questions

- Efficiency?

Numerical method: pseudo BVP in space–time domain

- Add dynamic pseudo boundary condition for long time boundary.
- Use adaptive wavelet multilevel solver with V-cycles for BVP.
- FAS approximation to cope with nonlinear equations.
- Iterate until residual satisfies L_2 norm tolerance.
- Split space–time domain in time direction into manageable slices.

Numerical method: pseudo BVP in space–time domain

- Add dynamic pseudo boundary condition for long time boundary.
- Use adaptive wavelet multilevel solver with V-cycles for BVP.
- FAS approximation to cope with nonlinear equations.
- Iterate until residual satisfies L_2 norm tolerance.
- Split space–time domain in time direction into manageable slices.

Numerical method: pseudo BVP in space–time domain

- Add dynamic pseudo boundary condition for long time boundary.
- Use adaptive wavelet multilevel solver with V-cycles for BVP.
- FAS approximation to cope with nonlinear equations.
- Iterate until residual satisfies L_2 norm tolerance.
- Split space–time domain in time direction into manageable slices.

Numerical method: pseudo BVP in space–time domain

- Add dynamic pseudo boundary condition for long time boundary.
- Use adaptive wavelet multilevel solver with V-cycles for BVP.
- FAS approximation to cope with nonlinear equations.
- Iterate until residual satisfies L_2 norm tolerance.
- Split space–time domain in time direction into manageable slices.

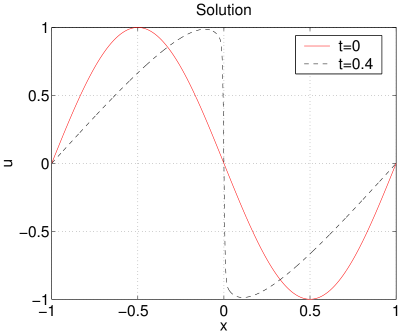
Numerical method: pseudo BVP in space–time domain

- Add dynamic pseudo boundary condition for long time boundary.
- Use adaptive wavelet multilevel solver with V-cycles for BVP.
- FAS approximation to cope with nonlinear equations.
- Iterate until residual satisfies L_2 norm tolerance.
- Split space–time domain in time direction into manageable slices.

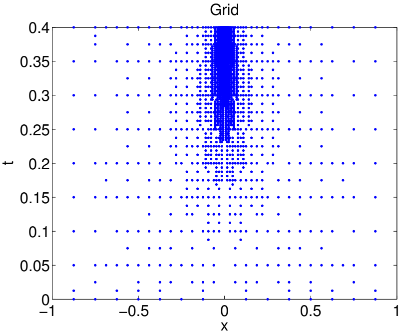
Numerical method: pseudo BVP in space–time domain

- Add dynamic pseudo boundary condition for long time boundary.
- Use adaptive wavelet multilevel solver with V-cycles for BVP.
- FAS approximation to cope with nonlinear equations.
- Iterate until residual satisfies L_2 norm tolerance.
- Split space–time domain in time direction into manageable slices.

Burgers equation: solution

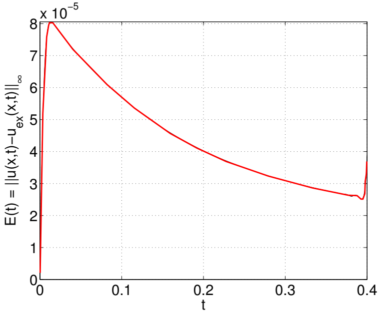


Solution

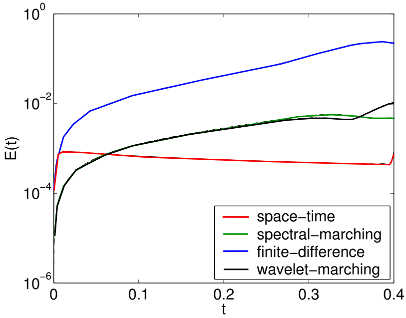


Adapted grid

Burgers equation: time integration error



Global error in time

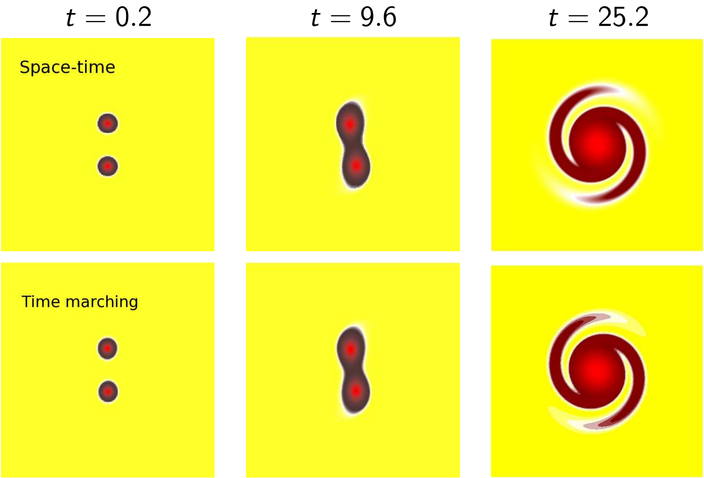


Comparison with time marching

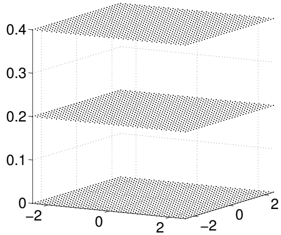
2D vortex merging

- Solve 2D vorticity equation for merger of identical vortices at $Re = 1\,000$.
- Use 2D+t domain of size $[-2.5, 2.5] \times [-2.5, 2.5] \times [0, 40]$.
- Total domain is divided into sub-domains of size $[-2.5, 2.5] \times [-2.5, 2.5] \times [0, 0.4]$.
- Four levels of refinement: max resolution in each subdomain is $256 \times 256 \times 16$.
- Solution converges after 5 iterations.

2D vortex merging: vorticity

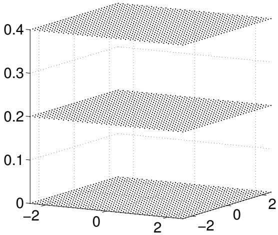


2D vortex merging: space–time grid

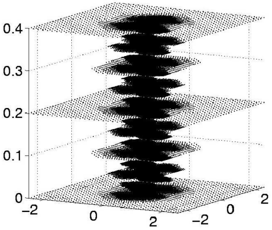


Initial grid
($t \in [0, 0.4]$)

2D vortex merging: space–time grid

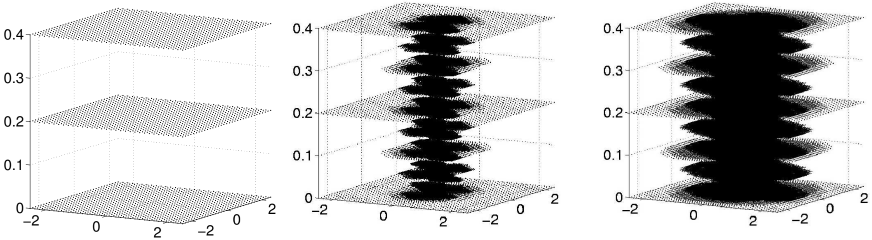


Initial grid
($t \in [0, 0.4]$)



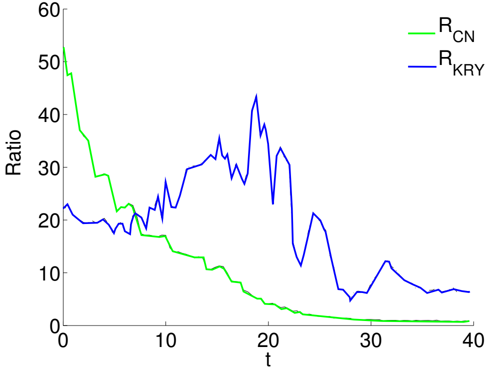
Converged grid
($t \in [0, 0.4]$)

2D vortex merging: space–time grid



Initial grid ($t \in [0, 0.4]$) Converged grid ($t \in [0, 0.4]$) Converged grid ($t \in [39.6, 40]$)

2D vortex merging: compression



$R_{\mathrm{CN}}(t)$: grid points used by the Crank-Nicolson time marching method compared to the number of grid points used by the space–time method in each sub-domain. $R_{\mathrm{KRY}}(t)$ is the equivalent ratio for the Krylov time marching method.

Stochastic Coherent LES

- If the threshold is small enough, i.e. $\epsilon \leq 10^{-3}$, the neglected modes are incoherent and need not be modelled.
- What if we want to use a very large threshold: $\epsilon = O(1)$?
- Effect of neglected subgrid scale modes on resolved modes can be modelled using a local form of the dynamic Smagorinsky model.
- This produces a dynamically adaptive form of large eddy simulation (LES) called SCALES.

Stochastic Coherent LES

- If the threshold is small enough, i.e. $\epsilon \leq 10^{-3}$, the neglected modes are incoherent and need not be modelled.
- What if we want to use a very large threshold: $\epsilon = O(1)$?
- Effect of neglected subgrid scale modes on resolved modes can be modelled using a local form of the dynamic Smagorinsky model.
- This produces a dynamically adaptive form of large eddy simulation (LES) called SCALES.

Stochastic Coherent LES

- If the threshold is small enough, i.e. $\epsilon \leq 10^{-3}$, the neglected modes are incoherent and need not be modelled.
- What if we want to use a very large threshold: $\epsilon = O(1)$?
- Effect of neglected subgrid scale modes on resolved modes can be modelled using a local form of the dynamic Smagorinsky model.
- This produces a dynamically adaptive form of large eddy simulation (LES) called SCALES.

Stochastic Coherent LES

- If the threshold is small enough, i.e. $\epsilon \leq 10^{-3}$, the neglected modes are incoherent and need not be modelled.
- What if we want to use a very large threshold: $\epsilon = O(1)$?
- Effect of neglected subgrid scale modes on resolved modes can be modelled using a local form of the dynamic Smagorinsky model.
- This produces a dynamically adaptive form of large eddy simulation (LES) called SCALES.

Stochastic Coherent LES

- If the threshold is small enough, i.e. $\epsilon \leq 10^{-3}$, the neglected modes are incoherent and need not be modelled.
- What if we want to use a very large threshold: $\epsilon = O(1)$?
- Effect of neglected subgrid scale modes on resolved modes can be modelled using a local form of the dynamic Smagorinsky model.
- This produces a dynamically adaptive form of large eddy simulation (LES) called SCALES.

Filtered SCALES equations

$$\frac{\partial \overline{u_i}^{>\epsilon}}{\partial t} + \frac{\partial(\overline{u_i}^{>\epsilon}\,\overline{u_j}^{>\epsilon})}{\partial x_j} = -\frac{1}{\rho}\frac{\partial \overline{p}^{>\epsilon}}{\partial x_i} + \nu \frac{\partial^2 \overline{u_i}^{>\epsilon}}{\partial x_j \partial x_j} - \frac{\partial \overline{\tau_{ij}}^{>\epsilon}}{\partial x_j},$$

$$\frac{\partial \overline{u_i}^{>\epsilon}}{\partial x_i} = 0,$$

where the wavelet filtering corresponding to a given threshold ϵ is denoted as $\overline{(\cdot)}^{>\epsilon}$, and $\overline{\tau_{ij}}^{>\epsilon}$ is the sub-grid scale stress to be modelled

$$\boxed{\overline{\tau_{ij}}^{>\epsilon} = \overline{u_i u_j}^{>\epsilon} - \overline{u_i}^{>\epsilon}\,\overline{u_j}^{>\epsilon}}$$

Sub-grid scale model

The sub-grid scale stress is modelled using the standard Smagorinsky eddy viscosity model

$$\boxed{\overline{\tau_{ij}}^{>\epsilon} = \nu_T \, \overline{S_{ij}}^{>\epsilon}}$$

where the eddy viscosity $\nu_T = -2C_S(t)\epsilon^2|\overline{S}^{>\epsilon}|$ and $\overline{S_{ij}}^{>\epsilon}$ is the strain rate of the resolved scales.

Sub-grid scale model

The sub-grid scale stress is modelled using the standard Smagorinsky eddy viscosity model

$$\boxed{\overline{\tau_{ij}}^{>\epsilon} = \nu_T \, \overline{S_{ij}}^{>\epsilon}}$$

where the eddy viscosity $\nu_T = -2C_S(t)\epsilon^2|\overline{S}^{>\epsilon}|$ and $\overline{S_{ij}}^{>\epsilon}$ is the strain rate of the resolved scales.

$C_S(t)$ is determined as in dynamic LES, by using the Germano identity and a test filter of twice the threshold, 2ϵ.

Advantages of SCALES compared to LES

- Most energetic part of coherent vortices is resolved at all scales, rather than just large scales.
- Filter scale adjusts automatically to account for flow inhomogeneity and intermittency.
- No special treatment of solid boundaries is required.

Advantages of SCALES compared to LES

- Most energetic part of coherent vortices is resolved at all scales, rather than just large scales.
- Filter scale adjusts automatically to account for flow inhomogeneity and intermittency.
- No special treatment of solid boundaries is required.

Advantages of SCALES compared to LES

- Most energetic part of coherent vortices is resolved at all scales, rather than just large scales.
- Filter scale adjusts automatically to account for flow inhomogeneity and intermittency.
- No special treatment of solid boundaries is required.

Advantages of SCALES compared to LES

- Most energetic part of coherent vortices is resolved at all scales, rather than just large scales.
- Filter scale adjusts automatically to account for flow inhomogeneity and intermittency.
- No special treatment of solid boundaries is required.

SCALES simulation of turbulence

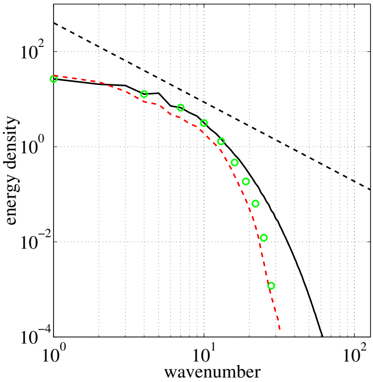
- Decaying homogeneous isotropic turbulence at $Re_\lambda = 72$.
- Maximum resolution is 256^3 (equivalent to pseudo-spectral 128^3).
- Initialized using de-aliased pseudo-spectral DNS.
- Threshold is $\epsilon = 0.5$.
- Compare to full wavelet 256^3 DNS and de-aliased 64^3 LES.

Results of SCALES for homogeneous isotropic turbulence

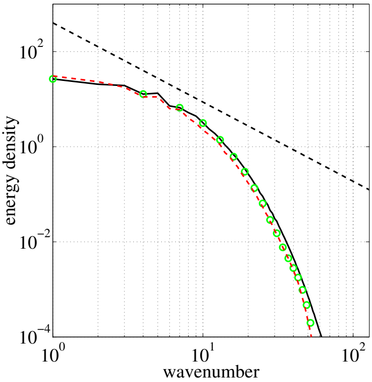
Vorticity isosurfaces at 30% $||\vec{\omega}||_\infty$

Only 1% of modes are resolved, i.e. 100 times compression

Comparison of energy spectra



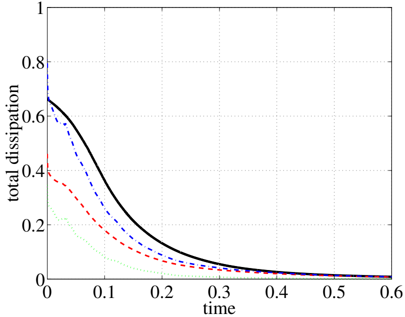
LES



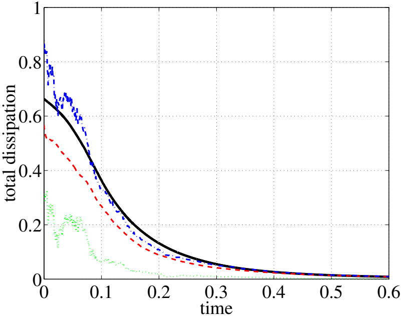
SCALES

Model energy spectrum compared to filtered DNS, and DNS.

Comparison of dissipation



LES



SCALES

Total model dissipation, viscous dissipation, SGS dissipation, DNS dissipation.

Summary

- Dynamic adaptivity is necessary for simulation of turbulence.
- Ideally, both spatial and temporal resolution should adapt to turbulence intermittency.
- Adaptive wavelet methods capture the dynamically important coherent vortices of the flow.
- The unresolved modes can be neglected (for small thresholds), or modelled simply (for high thresholds).

Summary

- Dynamic adaptivity is necessary for simulation of turbulence.
- Ideally, both spatial and temporal resolution should adapt to turbulence intermittency.
- Adaptive wavelet methods capture the dynamically important coherent vortices of the flow.
- The unresolved modes can be neglected (for small thresholds), or modelled simply (for high thresholds).

Summary

- Dynamic adaptivity is necessary for simulation of turbulence.
- Ideally, both spatial and temporal resolution should adapt to turbulence intermittency.
- Adaptive wavelet methods capture the dynamically important coherent vortices of the flow.
- The unresolved modes can be neglected (for small thresholds), or modelled simply (for high thresholds).

Summary

- Dynamic adaptivity is necessary for simulation of turbulence.
- Ideally, both spatial and temporal resolution should adapt to turbulence intermittency.
- Adaptive wavelet methods capture the dynamically important coherent vortices of the flow.
- The unresolved modes can be neglected (for small thresholds), or modelled simply (for high thresholds).

Summary

- Dynamic adaptivity is necessary for simulation of turbulence.
- Ideally, both spatial and temporal resolution should adapt to turbulence intermittency.
- Adaptive wavelet methods capture the dynamically important coherent vortices of the flow.
- The unresolved modes can be neglected (for small thresholds), or modelled simply (for high thresholds).

Summary

- Dynamic adaptivity is necessary for simulation of turbulence.
- Ideally, both spatial and temporal resolution should adapt to turbulence intermittency.
- Adaptive wavelet methods capture the dynamically important coherent vortices of the flow.
- The unresolved modes can be neglected (for small thresholds), or modelled simply (for high thresholds).

Successful simulation of high Reynolds number turbulence may lead to a true theory of turbulence.