# A review of mathematical regularization as model for small-scale turbulence


**Bernard J. Geurts**

**Multiscale Modeling and Simulation**


**Madrid, ICMAT, July 3-7**

# Turbulence in physical space

# Incompressible flow

Conservation of mass and momentum:

$$\partial_j u_j = 0$$

$$\partial_t u_i + \partial_j(u_i u_j) + \partial_i p - \frac{1}{Re}\partial_{jj} u_i = 0$$

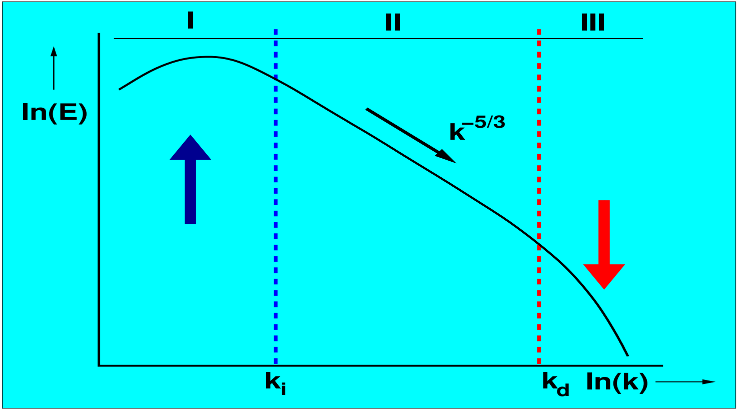
- convective flux: nonlinear, destabilizing
- viscous flux: linear, dissipative

Ratio – Reynolds number

$$Re = \frac{convection}{dissipation} = \frac{UL}{\nu}$$

Turbulence requires $Re \gg 1$

# Energy cascading process in 3D




**I:** large-scales stirring at integral length-scale $\ell_i \sim 1/k_i$

**II:** inviscid nonlinear transfer – inertial range $E \sim k^{-5/3}$

**III:** viscous dissipation dominant $\ell_d \sim 1/k_d$

# Complexity of turbulent flow

Kolmogorov: small scales are viscous, isotropic and universal
Proposition: properties depend on viscosity $\nu$ and dissipation rate $\varepsilon$

$$[\nu] = \text{length}^2/\text{time} \quad ; \quad [\varepsilon] = \text{length}^2/\text{time}^3$$

Scales: make length and time

$$\eta = \left(\frac{\nu^3}{\varepsilon}\right)^{1/4} \sim \frac{1}{k_d} \quad ; \quad \tau_\eta = \left(\frac{\nu}{\varepsilon}\right)^{1/2}$$

Three dimensions:

$$\#\text{dof} \sim \left(\frac{\ell}{\eta}\right)^3 \sim Re^{9/4} \quad ; \quad \#\text{time-steps} \sim \frac{t_{end}}{\tau_\eta} \sim Re^{1/2}$$

Computationally intensive problem as $Re \gg 1$

# Computational challenge

Reynolds scaling of numerical resolution

$$N = \left(\frac{\ell}{\eta}\right)^3 \sim Re^{9/4}$$

Memory: if $Re \rightarrow Re \times 10$ then $N \rightarrow 10^{9/4} \times N \approx 175 \times N$
Reynolds scaling of numerical work

$$Work \sim Re^{1/2}\ Re^{9/4} \sim\ Re^{11/4}$$

CPU: if $Re \rightarrow Re \times 10$ then $W \rightarrow 10^{11/4} W \approx 560\, W$
Tough simulation problem

- direct approach often impractical
- capture primary features instead

Four themes to be mastered:

- Phenomenology of (coarsened) turbulence
- Turbulence modeling and numerical methods
- Error-assessment for large-eddy simulation
- High-performance computing

Four themes to be mastered:

- Phenomenology of (coarsened) turbulence
- **Turbulence modeling and numerical methods**
- **Error-assessment for large-eddy simulation**
- High-performance computing

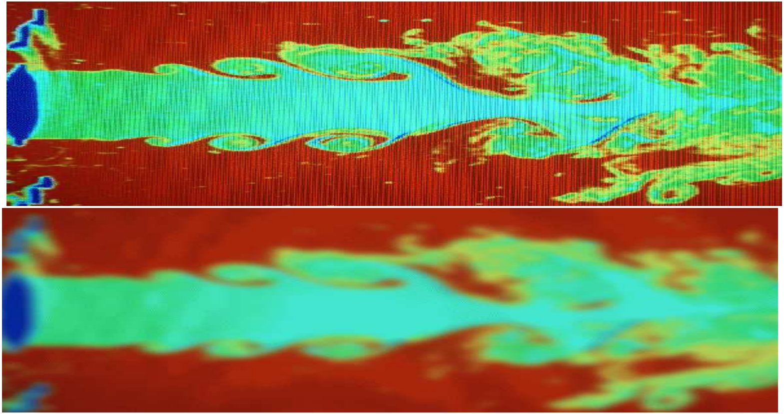
# Outline

1. Filering and closure
2. Regularized Navier-Stokes as turbulence model
3. Model testing
4. Concluding Remarks

# Outline

1. **Filtering and closure**
2. Regularized Navier-Stokes as turbulence model
3. Model testing
4. Concluding Remarks

# DNS and LES in a picture



- capture both large and small scales: resolution problem
- Coarsening/mathematical modeling instead: LES

# Filtering Navier-Stokes equations

$$\partial_j u_j = 0 \quad ; \quad \partial_t u_i + \partial_j(u_i u_j) + \partial_i p - \frac{1}{Re}\partial_{jj} u_i = 0$$

*Convolution-Filtering: filter-kernel G*

$$\overline{u}_i = L(u_i) = \int G(x-\xi)u(\xi)\,d\xi \quad ; \quad L(1) = 1$$

Large-eddy equations:

$$\partial_j \overline{u}_j = 0$$

$$\partial_t \overline{u}_i + \partial_j(\overline{u}_i \overline{u}_j) + \partial_i \overline{p} - \frac{1}{Re}\partial_{jj}\overline{u}_i = -\partial_j(\overline{u_i u_j} - \overline{u}_i \overline{u}_j)$$

Sub-filter stress tensor

$$\tau_{ij} = \overline{u_i u_j} - \overline{u}_i \overline{u}_j = L(\Pi_{ij}(\mathbf{u})) - \Pi_{ij}(L(\mathbf{u})) = [L, \Pi_{ij}](\mathbf{u})$$

# Spatial filtering, closure problem

After closure - Shorthand notation:

$$NS(\mathbf{u}) = 0 \quad \Rightarrow \quad NS(\overline{\mathbf{u}}) = -\nabla \cdot \tau(\mathbf{u}, \overline{\mathbf{u}}) \ \Leftarrow \ -\nabla \cdot M(\overline{\mathbf{u}})$$

Basic LES formulation

$$\textit{Find } v : \quad NS(v) = -\nabla \cdot M(v)$$

What models *M* are available/reasonable?

# Eddy-viscosity modeling

Obtain smoothing via increased dissipation:

$$\partial_t \overline{u}_i + \partial_j(\overline{u}_j \overline{u}_i) + \partial_i \overline{p} - \left(\frac{1}{Re} + \nu_t\right)\partial_{jj}\overline{u}_i = 0$$

Damp large gradients: dimensional analysis

$$\nu_t = \mathit{length} \;\times\; \mathit{velocity} \sim \Delta \;\times\; \Delta|\partial_x \overline{u}|$$

Effect: Strong damping at large filter-width and/or gradients

# Some explicit subgrid models

Popular models:

- Dissipation: Eddy-viscosity models, e.g., Smagorinsky

$$\tau_{ij} \rightarrow -\nu_t S_{ij} = -(C_S \Delta)^2 |S| S_{ij} \quad ; \quad \text{effect } \frac{1}{Re} \rightarrow \left( \frac{1}{Re} + \nu_t \right)$$

- Similarity: Inertial range, e.g., Bardina

$$\tau_{ij} \rightarrow [L, \Pi_{ij}](\overline{\mathbf{u}}) = \overline{\overline{u}_i \overline{u}_j} - \overline{\overline{u}}_i \overline{\overline{u}}_j$$

- Mixed models ?

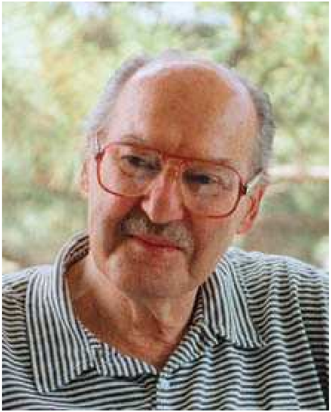
$$m_{ij} = Bardina + C_d Smagorinsky$$

$C_d$ via dynamic Germano-Lilly procedure

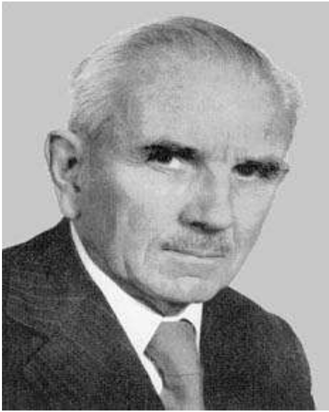
# Outline

1. Filtering and closure
2. Regularized Navier-Stokes as turbulence model
3. Model testing
4. Concluding Remarks

# Dissipation or regularization?



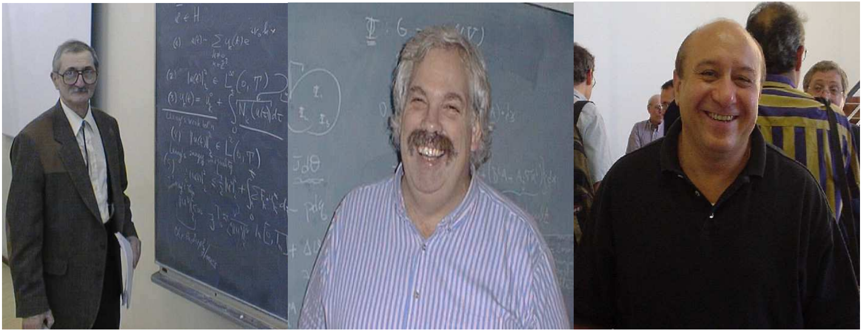
Smagorinsky



Leray

Capture turbulence with eddy-viscosity or, with mathematics?

# More regularization in LES



Ciprian Foias — Darryl Holm — Edriss Titi

Models with rigorously established existence, uniqueness, regularity, convergence to NS, transformation, symmetries, ...

# Mathematical regularization as subgrid model

Dynamic models popular in LES but:

- expensive; ad hoc implementation features ('clipping')
- still limited accuracy; complex flow extension difficult

Regularization principle directly altering the nonlinearity

- Systematically obtain implied subgrid closure
- 'Inherit' rigorous mathematical properties
- Maintain transport structure and transformation properties of equations

Consider two examples: Leray and NS-$\alpha$

# From Leray regularization to SFS model

Proposal:

$$\partial_t v_i + \overline{v}_j \partial_j v_i + \partial_i q - \frac{1}{Re}\partial_{jj} v_i = 0$$

Convolution filtering Leray: use $\partial_j \overline{v}_j = 0$

$$\partial_t \overline{v}_i + \partial_j(\overline{\overline{v}_j v_i}) + \partial_i \overline{q} - \frac{1}{Re}\partial_{jj}\overline{v}_i = 0$$

Rewrite into LES template:

$$\partial_t \overline{v}_i + \partial_j(\overline{v}_j \overline{v}_i) + \partial_i \overline{q} - \frac{1}{Re}\partial_{jj}\overline{v}_i = -\partial_j(\overline{\overline{v}_j v_i} - \overline{v}_j \overline{v}_i)$$

Implied Leray model:

$$m_{ij}^L = \overline{\overline{v}_j v_i} - \overline{v}_j \overline{v}_i = L\Big(\overline{v}_j L^{-1}(\overline{v}_i)\Big) - \overline{v}_j \overline{v}_i$$

with $L^{-1}$ formally inverting $L$

# Regularization requires filter-inversion

Geometric series: repeated filtering

$$L^{-1} = (I - (I - L))^{-1} \rightarrow \sum_{n=0}^{N} (I - L)^n$$

For example:

$$\begin{aligned}
N = 0: \quad u = &\quad L_0^{-1}(\overline{u}) = \overline{u} \\
N = 1: \quad u = &\quad L_1^{-1}(\overline{u}) = \overline{u} + (I - L)\overline{u} = 2\overline{u} - \overline{\overline{u}} \\
N = 2: \quad u = &\quad L_2^{-1}(\overline{u}) = \overline{u} + (I - L)\overline{u} + (I - L)(I - L)\overline{u} \\
= &\quad 3\overline{u} - 3\overline{\overline{u}} + \overline{\overline{\overline{u}}}
\end{aligned}$$

# Regularization requires filter-inversion

# Alternative regularizations

Consider $a - b$ models: $\partial_j v_j = 0$ and

$$\partial_t v_i + \partial_j(a_j b_i) + \partial_i q - \frac{1}{Re}\partial_{jj} v_i = 0$$

Then

- Choose: $a_j = v_j$; $b_i = v_i$ to obtain NS
- Choose: $a_j = \overline{v}_j$; $b_i = v_i$ to obtain Leray
- Choose: $a_j = v_j$; $b_i = \overline{v}_i$ to obtain modified Leray
- Choose: $a_j = \overline{v}_j$; $b_i = \overline{v}_i$ to obtain modified Bardina

Or, implied models: $m_{ij}^R = \overline{a_j b_i} - \overline{v}_j \overline{v}_i$, i.e.,

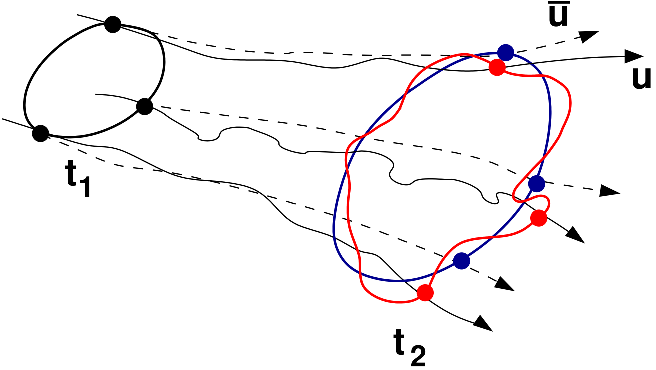
$$\begin{aligned}
\textit{Leray}: \quad & m_{ij}^{L} &= \overline{\overline{v}_j v_i} - \overline{v}_j \overline{v}_i \\
\textit{Modified Leray}: \quad & m_{ij}^{mL} &= \overline{v_j \overline{v}_i} - \overline{v}_j \overline{v}_i \\
\textit{Modified Bardina}: \quad & m_{ij}^{mB} &= \overline{\overline{v}_j \overline{v}_i} - \overline{v}_j \overline{v}_i
\end{aligned}$$

# NS-$\alpha$ regularization

## Kelvin's circulation theorem

$$\frac{d}{dt}\left(\oint_{\Gamma(\mathbf{u})} u_j \, dx_j\right) - \frac{1}{Re}\oint_{\Gamma(\mathbf{u})} \partial_{kk} u_j \, dx_j = 0 \quad \Rightarrow \quad NS - eqs$$

## Filtered Kelvin theorem ($\Gamma(\mathbf{u}) \to \Gamma(\overline{\mathbf{u}})$) extends Leray



# NS-$\alpha$ regularization

Filtered Kelvin circulation theorem

$$\frac{d}{dt}\left( \oint_{\Gamma(\overline{\mathbf{u}})} u_j \, dx_j \right) - \frac{1}{Re} \oint_{\Gamma(\overline{\mathbf{u}})} \partial_{kk} u_j \, dx_j = 0$$

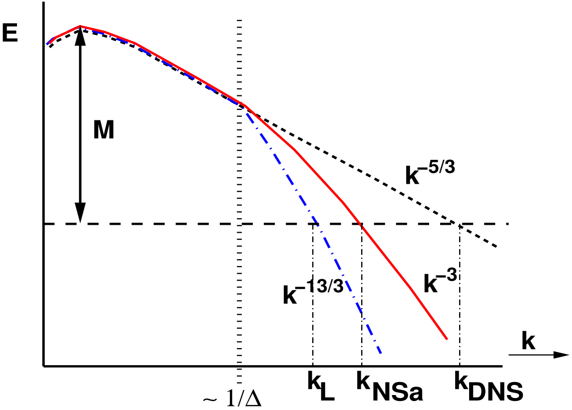
Euler-Poincaré

$$\partial_t u_j + \overline{u}_k \partial_k u_j + u_k \partial_j \overline{u}_k + \partial_j p - \partial_j (\frac{1}{2} \overline{u}_k u_k) - \frac{1}{Re} \partial_{kk} u_j = 0$$

Rewrite into LES template: Implied subgrid model

$$\begin{aligned} \partial_t \overline{u}_i \quad &+ \quad \partial_j (\overline{u}_j \overline{u}_i) + \partial_i \overline{p} - \frac{1}{Re} \partial_{jj} \overline{u}_i \\ &= \quad -\partial_j \left( \overline{\overline{u}_j u_i} - \overline{u}_j \overline{u}_i \right) - \frac{1}{2} \left( \overline{u_j \partial_i \overline{u}_j - \overline{u}_j \partial_i u_j} \right) \end{aligned}$$

# Cascade-dynamics – computability

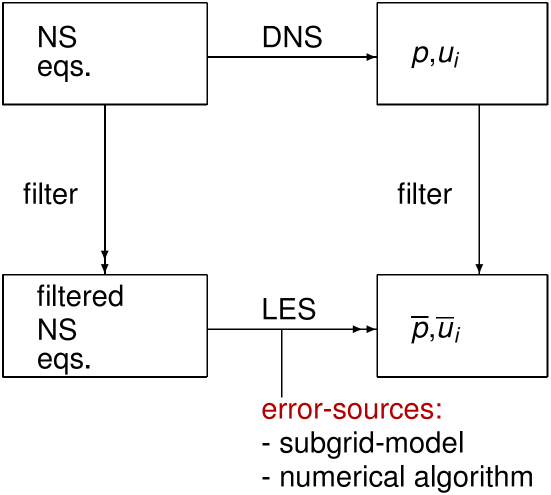



- NS-$\alpha$,Leray are dispersive
- Regularization alters spectrum – controllable cross-over as $k \sim 1/\Delta$: steeper than $-5/3$

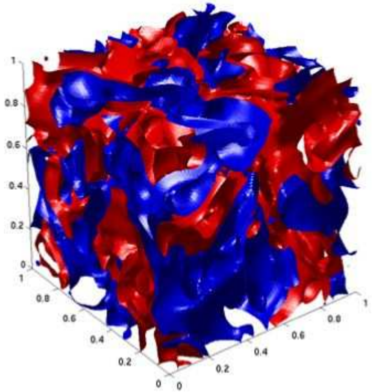
# Outline

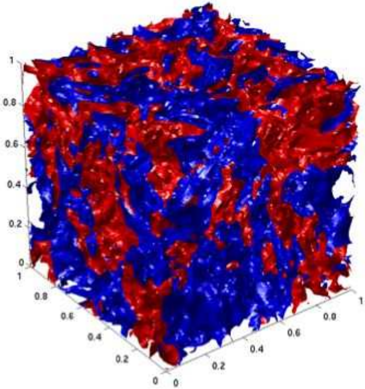
1. Filtering and closure
2. Regularized Navier-Stokes as turbulence model
3. **Model testing**
4. Concluding Remarks

# A posteriori testing



# HIT at $Re_\lambda = 50,\ 100$





Pseudo-spectral method and explicit time-stepping

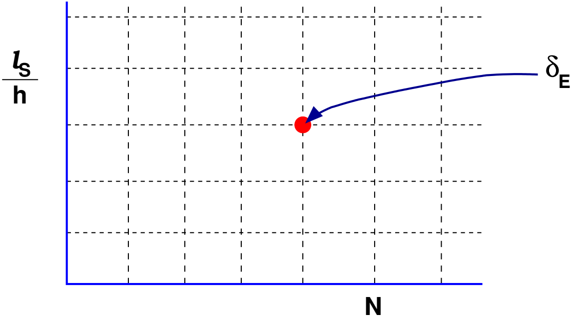
Range: $\Delta = \ell/64,\ \ell/32,\ \ell/16,\ \ell/8,\ \ell/4,\ \ell/2$ at

$N = 32, 64, 128, 256, 512$

# Smagorinsky

# Error-landscape: Definition

Framework for collecting error information:



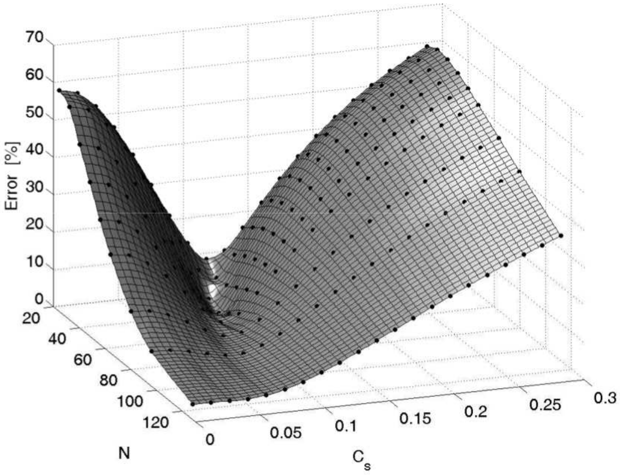

Each Smagorinsky LES corresponds to single point:

$$\left(N, \frac{\ell_S}{h}\right) \quad ; \quad error : \quad \delta_E$$

Contours of energy error $\delta_E$ — fingerprint of LES

# Interacting simulation errors

Computational model limited by numerical and modeling errors

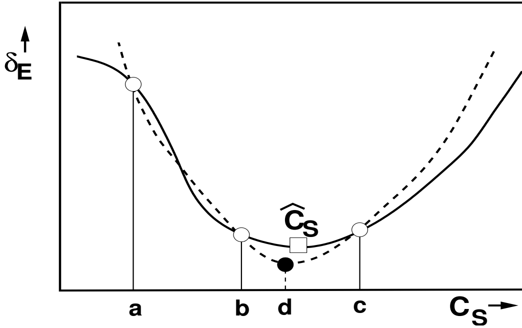



HIT error landscape

Iterative optimization?

# SIPI - basic algorithm

Goal: minimize total error at given $N$



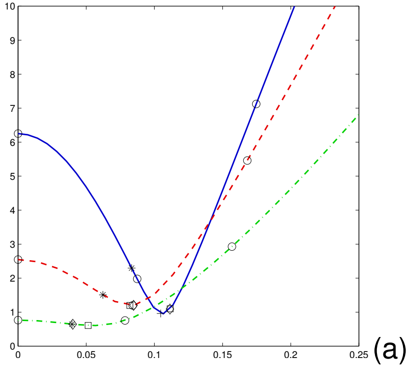

Initial triplet: no-model, dynamic and half-way

New iterand

$$d = b - \frac{1}{2} \frac{(b-a)^2[\delta_E(b) - \delta_E(c)] - (b-c)^2[\delta_E(b) - \delta_E(a)]}{(b-a)[\delta_E(b) - \delta_E(c)] - (b-c)[\delta_E(b) - \delta_E(a)]}$$

# SIPI applied to homogeneous turbulence

Each iteration = separate simulation

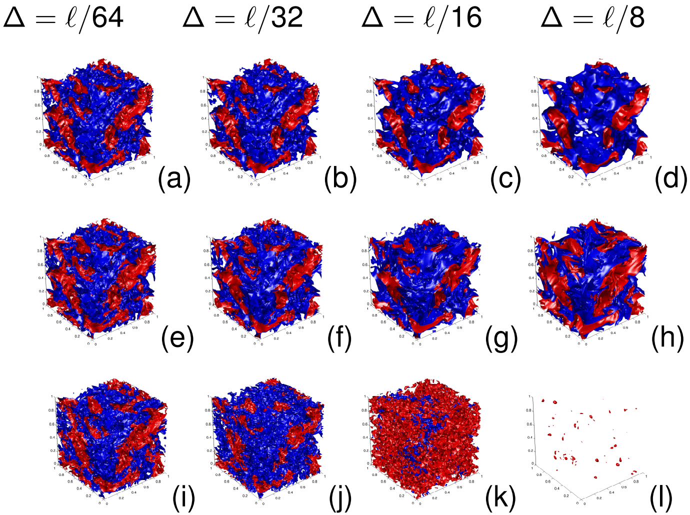



$Re_\lambda = 50$ (a) and $Re_\lambda = 100$ (b). Resolutions $N = 24$ (solid), $N = 32$ (dashed) and $N = 48$ (dash-dotted)

Iterations: $\circ \rightarrow * \rightarrow \diamond \rightarrow \Box \rightarrow +$

# Regularization

# Flow-structures: DNS, Leray, modified Leray




DNS at $512^3$ and LES at $128^3$

# Skewness prediction

## Grid-independent LES: $N = 128$

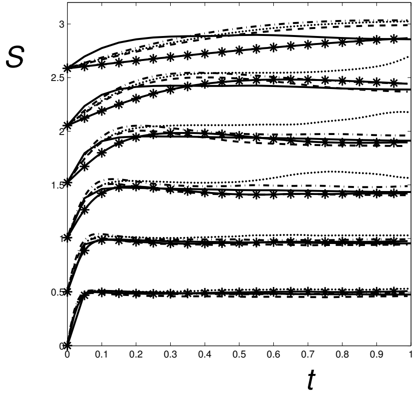



**Figure:** Filtered DNS (solid), Leray (dash), NS-$\alpha$ (dash-dot), Modified Leray (dot) and Modified Bardina (solid with $*$). From bottom to top: $\Delta = \ell/64, \ell/32, \ell/16, \ell/8, \ell/4$ and $\Delta = \ell/2$ – curves are shifted

# Numerical contamination Leray: $Re_\lambda$

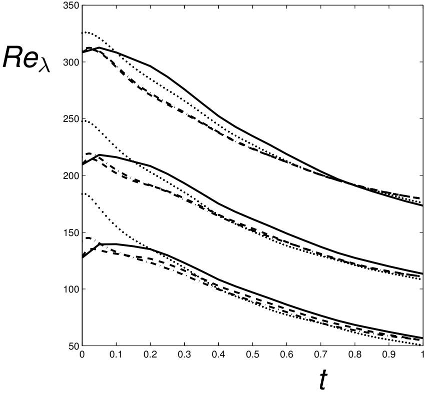



**Figure:** $N = 128$ (dash), $N = 64$ (dash-dot), $N = 32$ (dot) and $N = 16$ (solid) (bottom to top) $\Delta = \ell/64$, $\ell/32$ and $\ell/16$.

# Is regularization modeling practical?

Computational speed-up

$$W \approx \left(N_{DNS}/N_{LES}\right)^4$$

Increased *Re*: factor $\approx W^{1/3}$ since complexity $\sim Re^3$

General impression:

- 'very accurate' predictions at small filter-widths: $\Delta/\ell \lesssim 1/64$, requiring $N \approx 128$

  $W \approx 256$ *allows factor $\approx 6$ in Re*

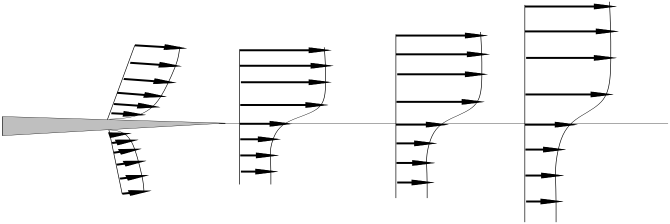
- 'quite accurate' predictions as $1/32 \lesssim \Delta/\ell \lesssim 1/16$, requiring $N = 32$ to $N = 64$ provided proper SFS model

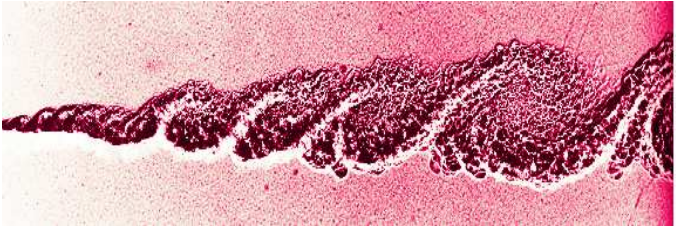
  $W \approx 4096 - 65536$ *allows factor $\approx 16$ to $\approx 40$ in Re*

- 'sometimes still OK' as $\Delta/\ell \approx 1/8$, requiring $N \approx 16$: $W \approx 10^6$, i.e., factor $\approx 100$ in *Re*
- considerable errors at very large filter-widths: $\Delta/\ell \gtrsim 1/4$

# Mixing

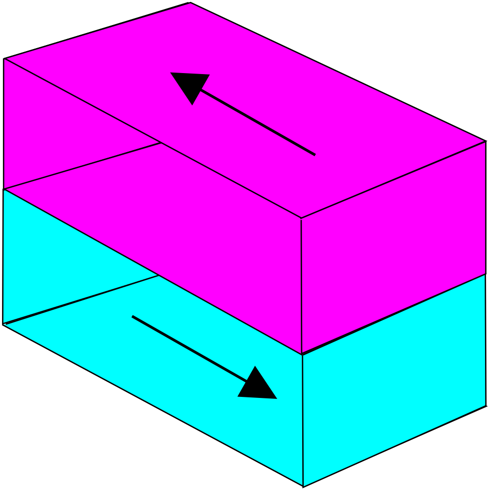
# Mixing layer: testing ground for LES

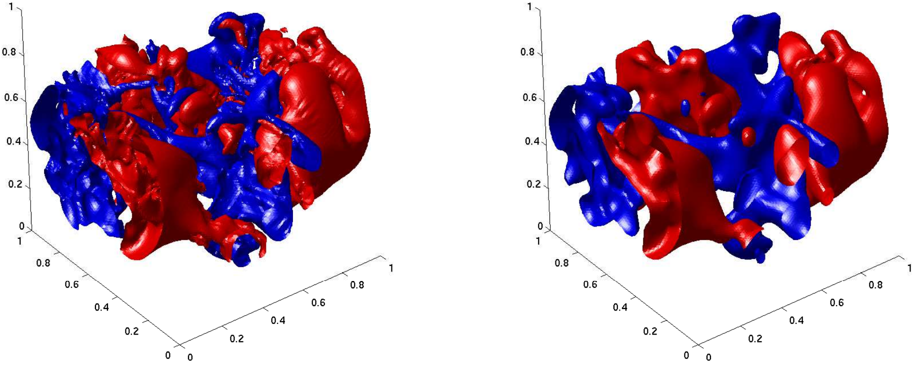

(a)


(b)

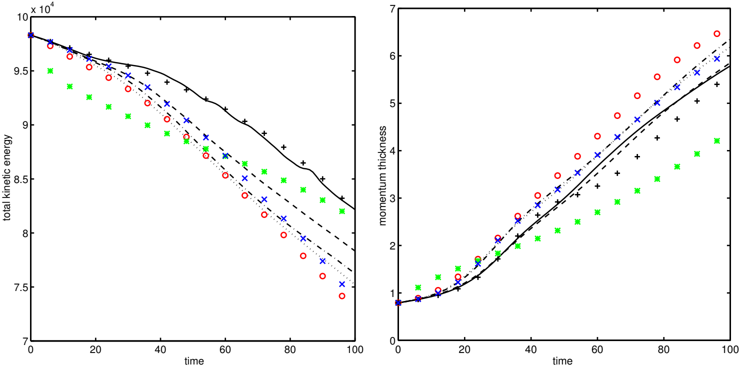
(a): Flow domain mixing layer
(b): Spark shadow photograph

# Basic mixing layer configuration




0
0.2
0.4
0.6
0.8
1
0
0.2
0.4
0.6
0.8
1
0
0.2
0.4
0.6
0.8
1
0
0.2
0.4
0.6
0.8
1
0
0.2
0.4
0.6
0.8
1
0
0.2
0.4
0.6
0.8
1

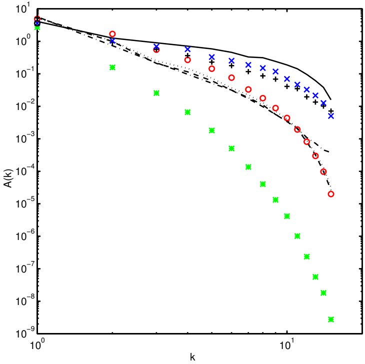
# Some mean flow properties




Kinetic energy and momentum thickness

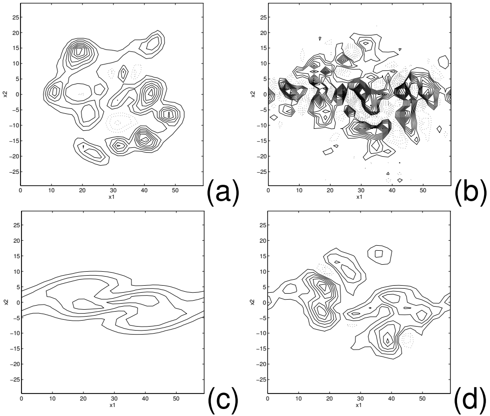
- Smagorinsky too dissipative
- Bardina, dynamic models preferred

# Closer look: Streamwise energy spectrum




- Dissipation: Smagorinsky too much, Bardina not enough
- dynamic models quite acceptable
- ‘middle range’ wavenumbers much too high

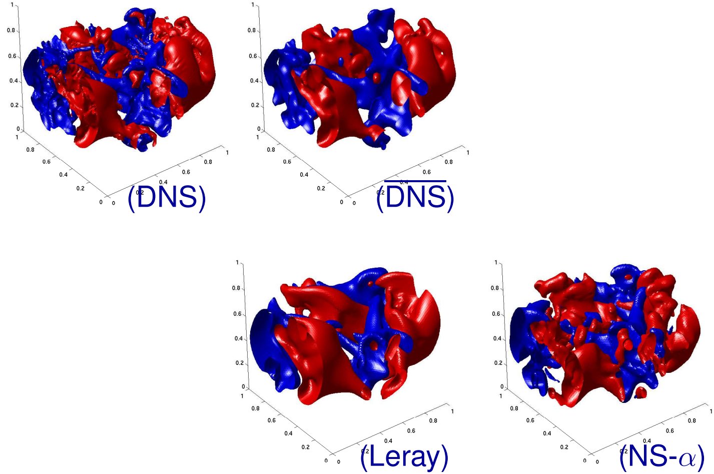
# Instantaneous snapshots of spanwise vorticity




- *a: $\overline{DNS}$, b: Bardina, c: Smagorinsky, d: dynamic*

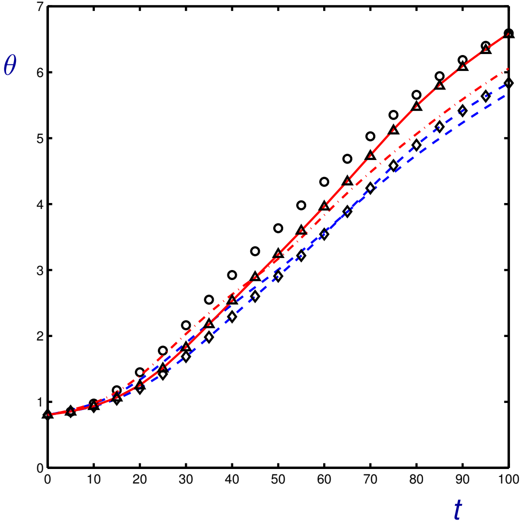
  Accuracy limited: regularization models better?

# Leray and NS-$\alpha$ predictions: $Re = 50$, $\Delta = \ell/16$




Snapshot $u_2$: red (blue) corresponds to up/down

# Momentum thickness $\theta$ as $\Delta = \ell/16$




**Filtered DNS (○)**

**Leray-model**

$32^3$: **dash-dotted**

$64^3$: **solid**

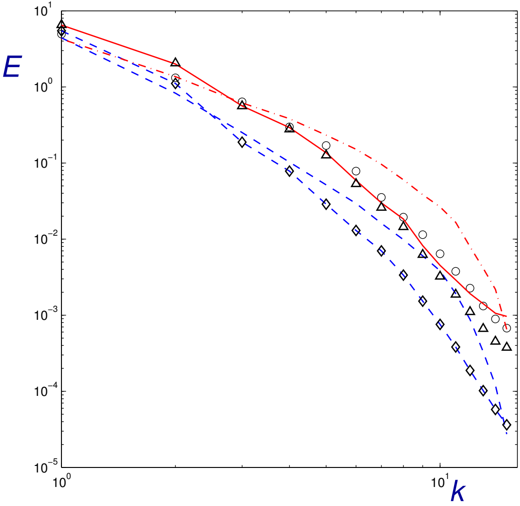
$96^3$: $\triangle$

**dynamic model**

$32^3$: **dashed**

$64^3$: **dashed with** $\diamond$

# Streamwise kinetic energy $E$ as $\Delta = \ell/16$




**Filtered DNS ($\circ$)**

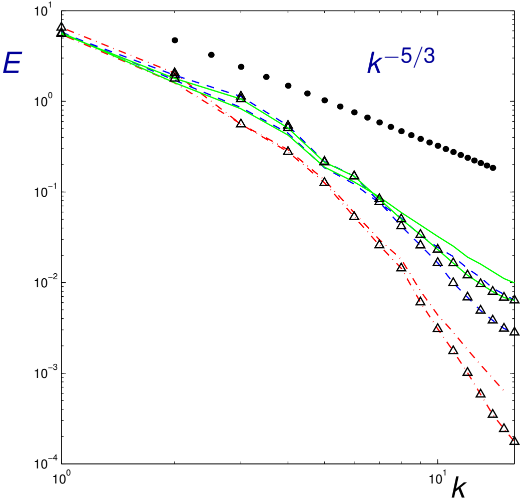
**Leray-model**

- $32^3$: **dash-dotted**
- $64^3$: **solid**
- $96^3$: $\triangle$

**dynamic model**

- $32^3$: **dashed**
- $64^3$: **dashed with** $\diamond$

# Robustness at arbitrary Reynolds number




$Re = 50$

$64^3$: **dash-dotted**

$96^3$: **dash-dotted,** $\triangle$

$Re = 500$

$64^3$: **dashed**

$96^3$: **dashed,** $\triangle$

$Re = 5000$

$64^3$: **solid**

$96^3$: **solid, =**$\triangle$

# Outline

1. Filtering and closure
2. Regularized Navier-Stokes as turbulence model
3. Model testing
4. **Concluding Remarks**

# Concluding remarks

Does mathematical regularization imply accurate SFS model?

- reviewed coarsened turbulence
- closure problem: eddy-viscosity and regularization
- illustrated a posteriori testing for HIT and mixing layer
- Leray and NS-$\alpha$ are accurate and Leray is more robust

- open challenge:
  - what fluid-mechanical properties should be included for successful NS regularization?
  - what is needed to assure/predict simulation reliability?