

Are adaptive wavelet discretizations dissipative?

Applications to inviscid Burgers and incompressible Euler equations

Kai Schneider

Aix-Marseille Université, France

in collaboration with

**Rodrigo Pereira¹, Romain Nguyen van yen², Marie Farge³
Naoya Okamoto⁴ and Katsunori Yoshimatsu⁴**

¹UFRJ Rio de Janeiro, Brazil, ²FU Berlin, Germany, ³ENS Paris, France, ⁴Nagoya University, Japan

Workshop: Turbulent Dissipation, Mixing and Predictability
UCLA Institute of Pure and Applied Mathematics
January 9 -13, 2017, Los Angeles, CA, USA

Outline

- Math. Framework: Dynamical Galerkin schemes
- Wavelet regularization of inviscid PDEs
 - 1D Burgers
 - 2D incompressible Euler
 - 3D incompressible Euler
- Conclusion
- (A characteristic mapping method for 2D Euler)

1. Dynamical Galerkin schemes (1)

2.1. Formal definition. Let H be a Banach space, and consider the equation

$$(2.1) \quad u' = f(u)$$

where u' denotes the weak time derivative of u , where f is defined and continuous from some sub-Banach space $D(f) \subset H$ into H .

Below we shall focus on the case of the 1D Burgers equation on the torus $\mathbb{T} = \mathbb{R}/\mathbb{Z}$:

$$(2.2) \quad \partial_t u + u \partial_x u = \nu \partial_{xx} u$$

which corresponds to (2.1) with

$$(2.3) \quad f(u) = \nu \partial_{xx} u - u \partial_x u$$

The classical Galerkin discretization of (2.1) is defined as follows: for $h > 0$, let H_h be a fixed finite dimensional subspace of $D(f)$, such that:

$$\overline{\bigcup_{h>0} H_h} = H$$

where the adherence is taken in H , and let P_h be the orthogonal projector on H_h .

1. Dynamical Galerkin schemes (2)

Find $u_h : [0, T] \in H_h$ such that:

$$(2.4) \quad u'_h = P_h f(u_h)$$

For $t \in [0, T]$, $P_h(t)$ is an orthogonal projector on a finite dimensional subspace $H_h(t)$ of H . The dimension of $H_h(t)$ is allowed to change in time, but $H_h(t)$ remains in H_h^0 . P_h therefore takes its values in the set of orthogonal projectors $H_h^0 \rightarrow H_h^0$, which we denote by Π_h^0 . We want to find $u_h : [0, T] \in H_h(t)$ which is an approximation of u .

First we assume that P_h is smooth in time. Thus we get

$$(2.5) \quad P_h(t)u'_h(t) = P_h(t)f(u_h(t))$$

but now, since P_h does not commute with the time-derivative, this equation is not sufficient to determine $u'_h(t)$ entirely. We need another equation to fix the component of $u'_h(t)$ which is in the orthogonal of $H_h(t)$.

1. Dynamical Galerkin schemes (3)

To derive this equation, we start with $u_h(t) \in H_h(t)$ for every t , which is equivalent to

$$(2.6) \quad P_h(t)u_h(t) = u_h(t).$$

Differentiating in time leads to:

$$(2.7) \quad P_h(t)u'_h(t) + P'_h(t)u_h = u'_h(t)$$

or equivalently

$$(2.8) \quad (1 - P_h(t))u'_h(t) = P'_h(t)u_h$$

which is the equation we were looking for. By adding (2.5) and (2.8) together, we obtain the definition of the dynamical Galerkin scheme:

$$(2.9) \quad u'_h(t) = P_h(t)f(u_h(t)) + P'_h(t)u_h(t)$$

By comparing this differential equation with (2.4), we observe the appearance of a new term proportionnal to the time-derivative of P_h . This is the essential ingredient which characterizes the dynamical Galerkin scheme.

1. Dynamical Galerkin schemes (4)

LEMMA 2.1. *Any solution of (2.9) such that $u_h(0) \in H_h(0)$ also satisfies $u_h(t) \in H_h(t)$ for all t , and moreover*

$$(2.10) \quad \frac{1}{2} \frac{d}{dt} \|u_h(t)\|^2 = (u_h(t), f(u_h(t)))$$

Proof. By differentiating $P_h(t)^2 = P_h(t)$ and $P_h(t)^3 = P_h(t)$ respectively, we obtain the identities

$$P_h(t)P_h(t)' + P_h(t)'P_h(t) = P_h(t)' \quad \text{and} \quad P_h(t)P_h(t)'P_h(t) = 0,$$

which imply that

$$(2.11) \quad \frac{d}{dt} ((1 - P_h(t))u_h(t)) = 0$$

and the first part follows. To prove the second part, take the inner product of the equation with u_h :

$$(2.12) \quad \frac{1}{2} \frac{d}{dt} \|u_h(t)\|^2 = (u_h(t), f(u_h(t))) + (u_h(t), P_h'(t)u_h(t)),$$

where the last term can be rewritten

$$(P_h(t)u_h(t), P_h'(t)P_h(t)u_h(t)) = (u_h(t), P_h(t)P_h'(t)P_h(t)u_h(t)) = 0.$$

1. Dynamical Galerkin schemes (5)

The above computations are valid when P_h is differentiable, which is a severe restriction and forbids us in particular to switch on and off dynamically some functions in the basis of integration. To pursue we therefore need to extend the definition of the scheme to non-differentiable P_h . For this we consider the integral formulation of (2.9), namely

$$(2.13) \quad u_h(t) = u_h(0) + \int_0^t P_h(\tau) f(u_h(\tau)) d\tau + \int_0^t P_h'(\tau) u_h(\tau) d\tau.$$

This equation can be rewritten using a Stieltjes integral with respect to P_h :

$$(2.14) \quad u_h(t) = u_h(0) + \int_0^t P_h(\tau) f(u_h(\tau)) d\tau + \int_0^t dP_h(\tau) u_h(\tau)$$

which we call the integral formulation of the dynamical Galerkin scheme.

This equation makes sense as soon as P_h has bounded variation, which gives it a much wider range of applicability than (2.9), allowing in particular discontinuities in P_h . To solve such an equation we need to resort to the theory of generalized ordinary differential equations.

1. Dynamical Galerkin schemes (6)

2.2. Existence and uniqueness of a solution to the projected equations.

The rigorous setting for integral equations such as (2.14) involving Stieltjes integrals is explained in detail in the book [9]. An alternative introduction can be found in [3]. We summarize the main consequences of the theory for our problem in the following

THEOREM 2.2. *Assume that $P_h(t) : [0, T] \rightarrow$ is BV and left-continuous, that $P_h(0)u_h(0) = u_h(0)$ (i.e. $u_h(0) \in H_h(0)$), and that $f : H_h^0 \rightarrow H$ is locally Lipschitz. Then*

(i) *There exists T^* , $0 < T^* \leq T$, such that the integral equation*

$$(2.15) \quad u_h(t) = u_h(0) + \int_0^t P_h(\tau) f(u_h(\tau)) d\tau + \int_0^t dP_h(\tau) u_h(\tau)$$

has a unique BV, left-continuous solution $u_h : [0, T^] \rightarrow H_h^0$.*

(ii) *This solution satisfies*

$$(2.16) \quad \forall t \in [0, T], P_h(t)u_h(t) = u_h(t)$$

1. Dynamical Galerkin schemes (7)

(iii) u_h is continuous at any point of continuity of P_h , and more generally for any t :

$$(2.17) \quad u_h(t^+) - u_h(t) = (P_h(t^+) - P_h(t))u_h(t)$$

or equivalently

$$(2.18) \quad u_h(t^+) = P_h(t^+)u_h(t)$$

(iv) The energy equation (2.10) for smooth P_h is replaced in general by:

$$(2.19) \quad \frac{1}{2}(\|u_h(t)\|^2 - \|u_h(0)\|^2) = \int_0^t (u_h(\tau), f(u_h(\tau)))d\tau - \frac{1}{2} \sum_{\{i|t_i < t\}} \|(1 - P_h(t_i^+))u_h(t_i)\|^2,$$

where $(t_i)_{i \in \mathbb{N}}$ are the points of discontinuity of P_h .

Proof: Schauder-Tichonov fixed point theorem and theorems from ref.
[9] S. Schwabik. Generalized ODEs. World Scientific, 1992.

2. Wavelet representation

$$\psi_{ji}(x) = 2^{j/2} \psi(2^j x - i)$$

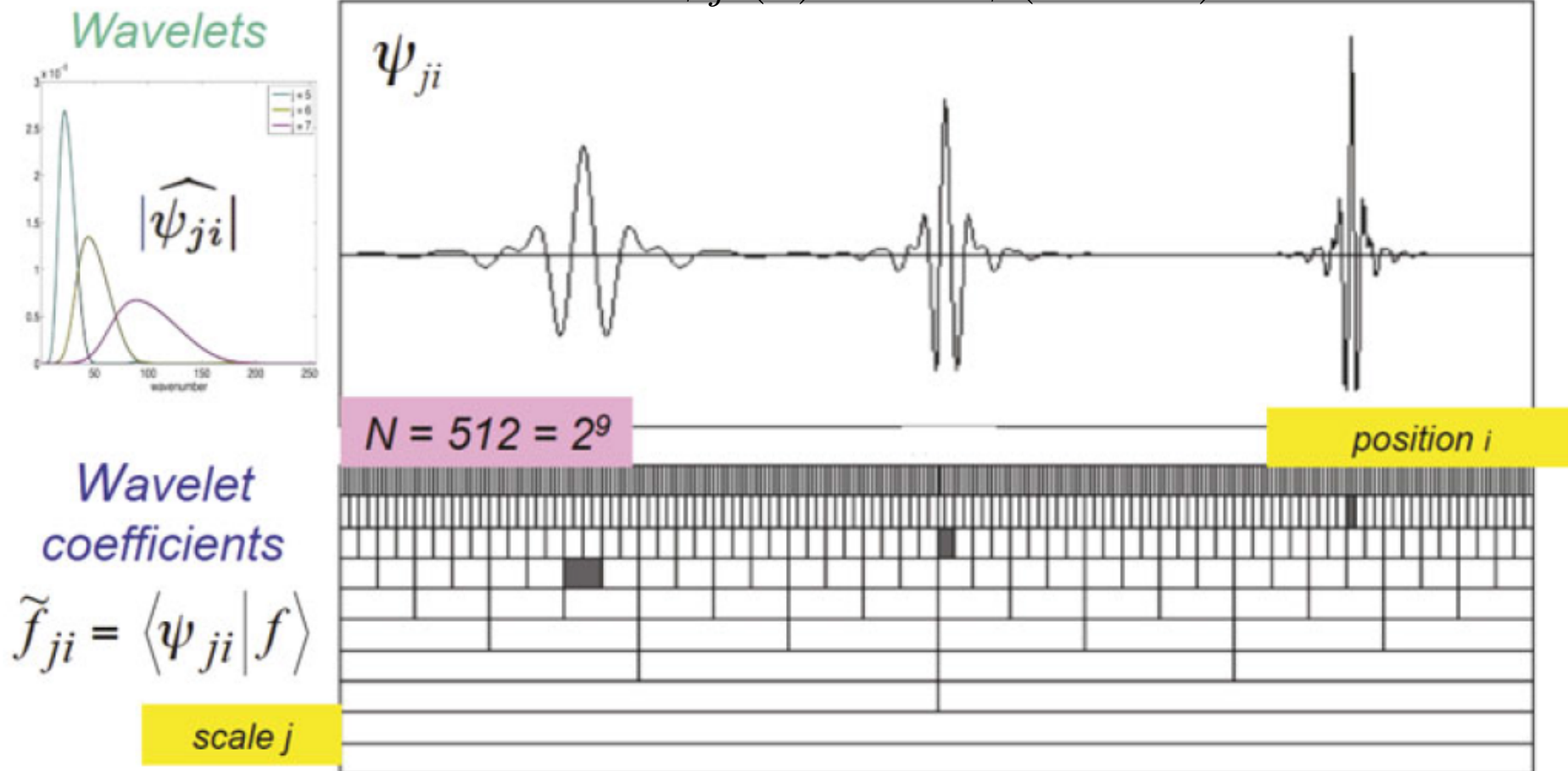


FIGURE 2. Space-scale representation of an orthogonal spline wavelet at three different scales and positions, i.e. $\psi_{6,6}$, $\psi_{7,32}$, $\psi_{8,108}$. The modulus of the Fourier transform of three corresponding wavelets is shown in the inset (top, left).

2. Applications

1. Comparison between
viscous, inviscid and wavelet-regularized
Galerkin-truncated 1D Burgers

2. Comparison between
2D Navier-Stokes, 2D Euler
and wavelet-regularized 2D Euler

3. Comparison between
3D Navier-Stokes, 3D Euler
and wavelet-regularized 3D Euler

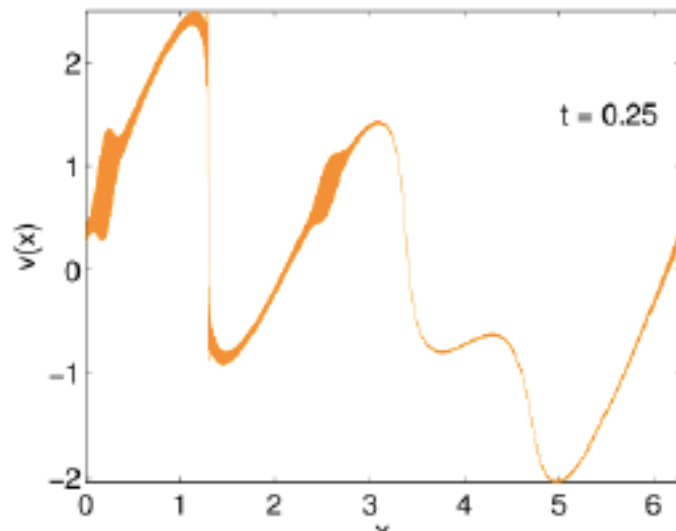
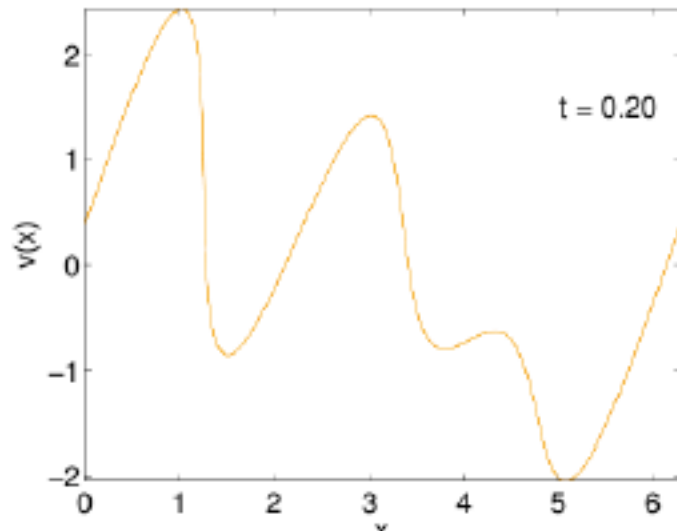
Galerkin-truncated 1D Burgers

- Inviscid Burgers equation

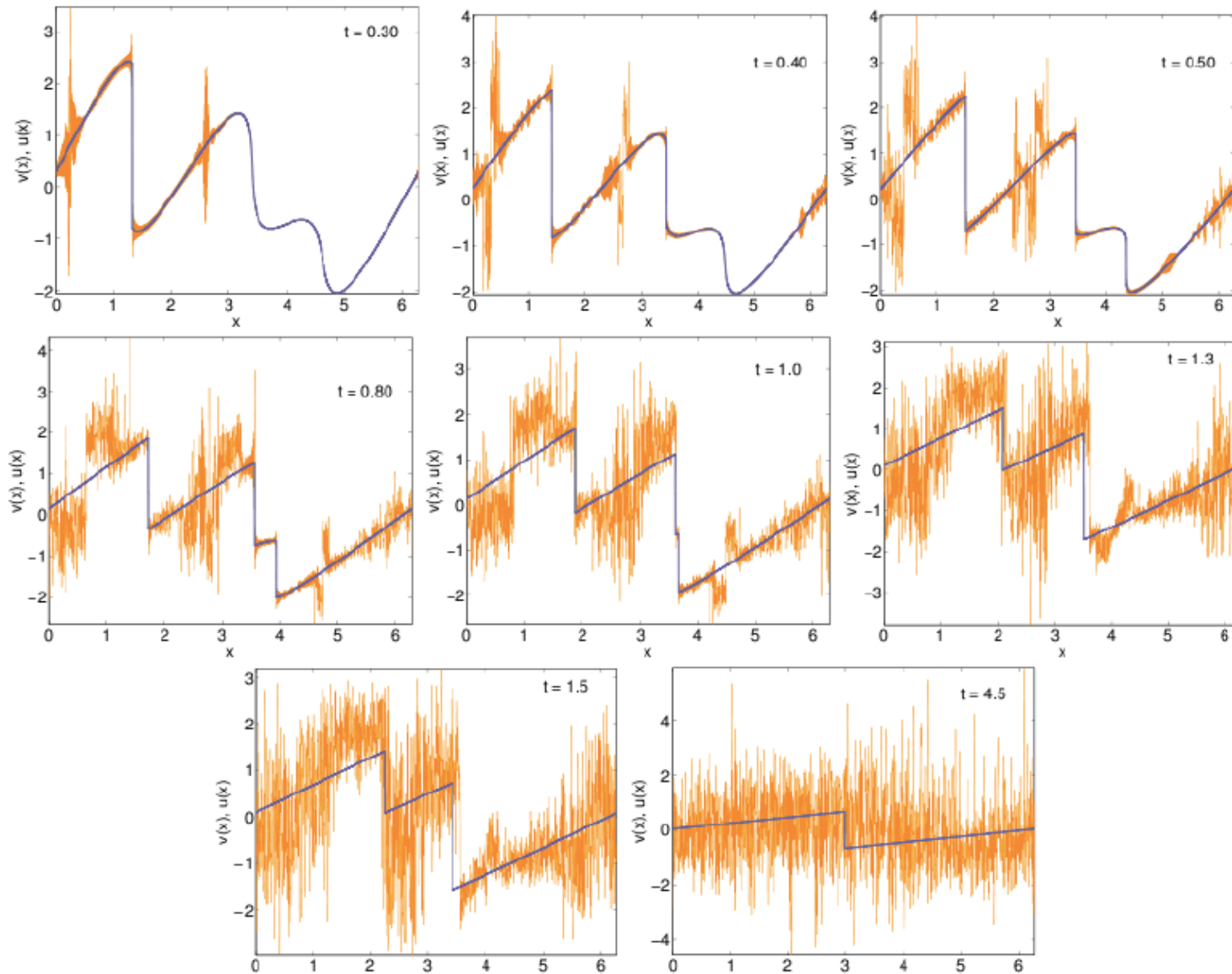
$$\partial_t u + \partial_x (u^2/2) = 0; \quad u(x, 0) = u_0(x)$$
$$u_0(x) = \sin(2\pi x) + \sin(4\pi x + 0.9) + \sin(6\pi x)$$

- Galerkin projector

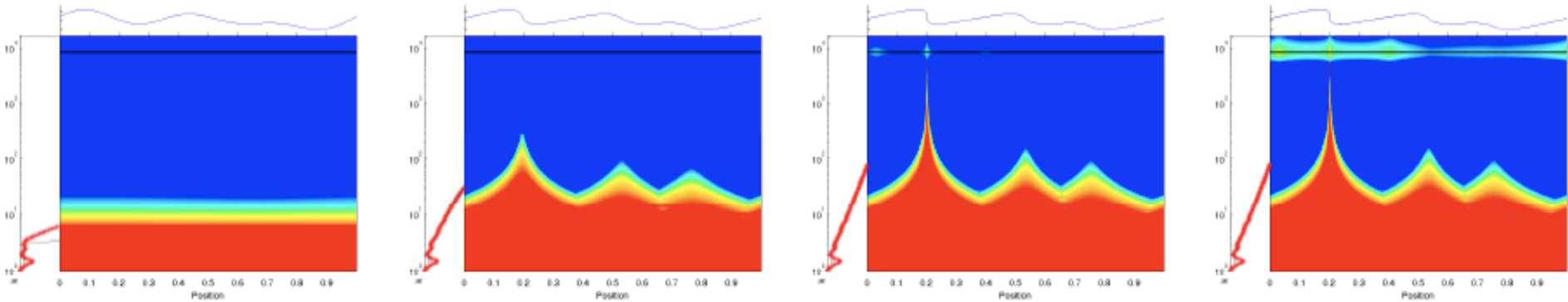
$$P_{K_G} u(x) = \sum_{|k| \leq K_G} e^{ikx} \hat{u}_k$$



Galerkin-truncated / analytic solution



Continuous wavelet analysis

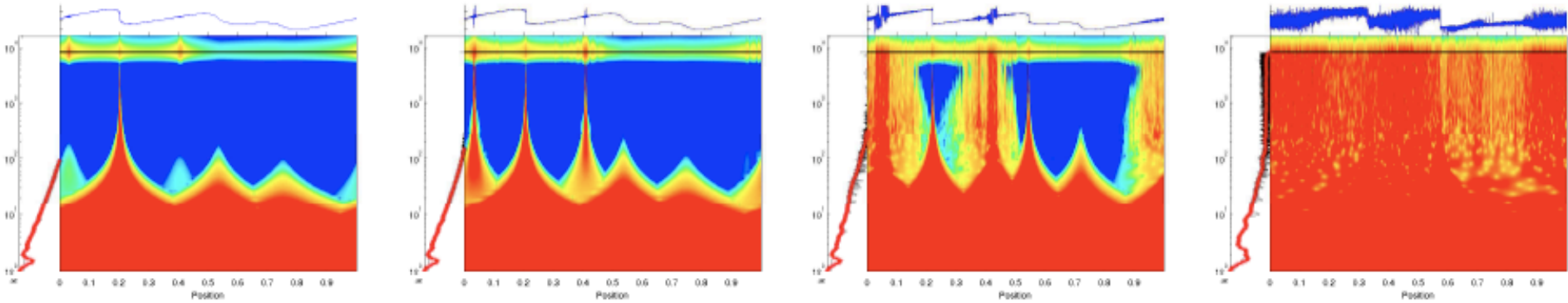


$t = 0$

$t = 0.02749$

$t = 0.03505$

$t = 0.03538$

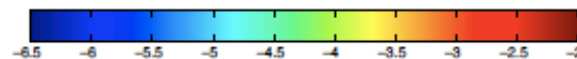


$t = 0.03648$

$t = 0.03998$

$t = 0.05897$

$t = 0.19989$



Wavelet regularization

- Wavelet-based method to select relevant degrees of freedom
- Well fit to turbulence since local irregularities won't disturb all coefficients
- Aim: track coherent structures
- But what is a coherent structure? Few would agree on that.
- Though few would disagree that noise certainly isn't.
- CVS → denoising in wavelet space
- Iterative scheme to define the “incoherent” part of the signal, to be discarded
- Only coherent modes are evolved

*Farge, Schneider & Kevlahan,
Phys. Fluids, 2(8), 2187, 1999*

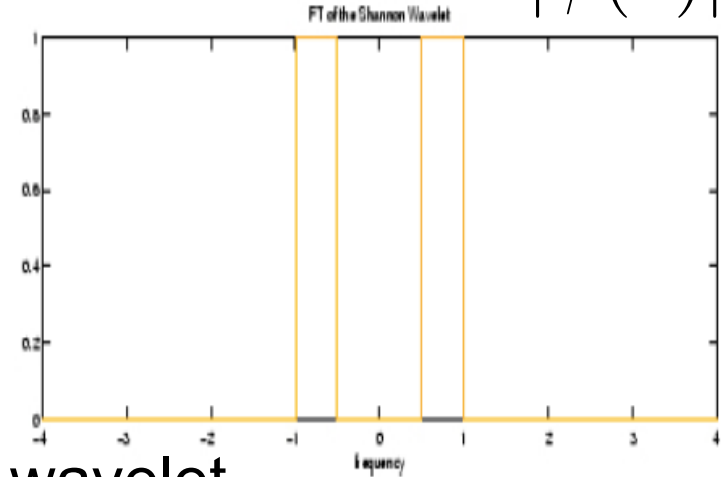
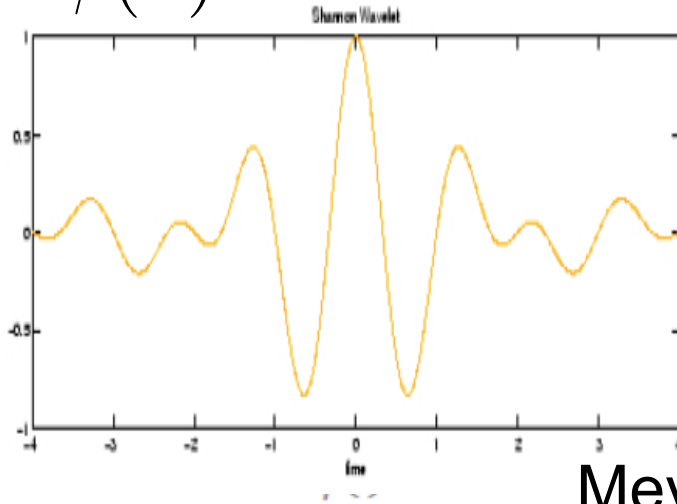
*Nguyen, Farge, Kolomenskiy, Schneider
& Kingsbury, Physica D, 237, 2008*

Real orthogonal wavelets

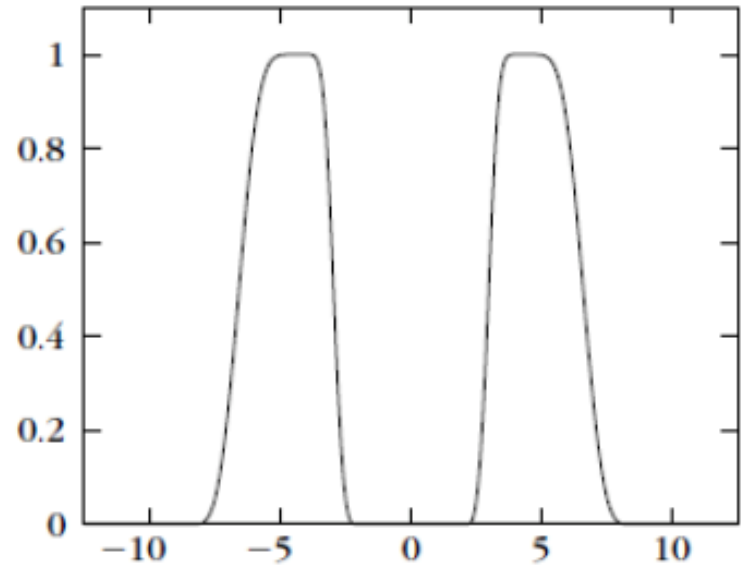
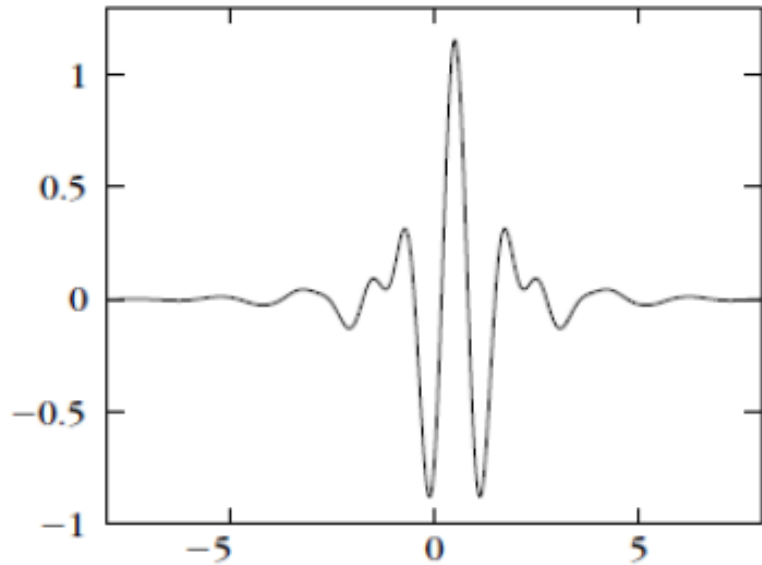
$\psi(x)$

Shannon wavelet

$|\hat{\psi}(k)|$

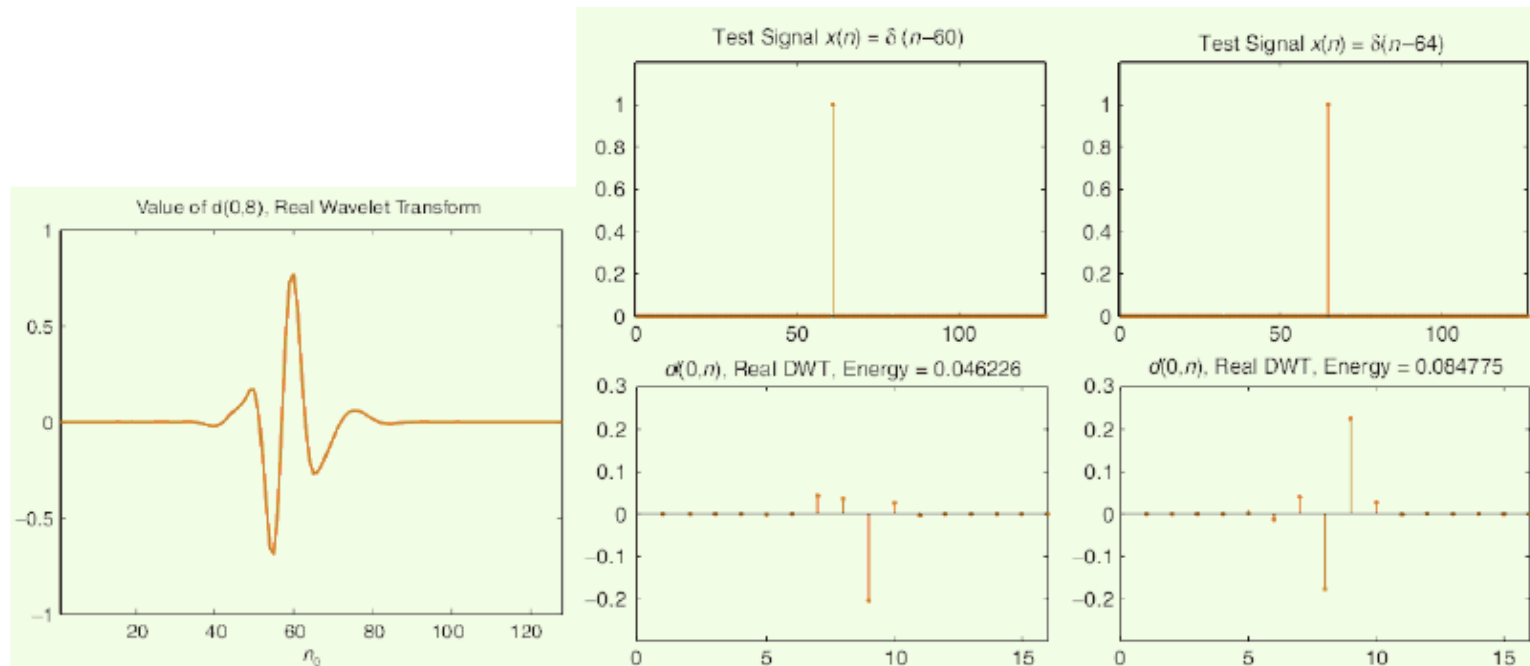


Meyer wavelet



Lack of translational invariance

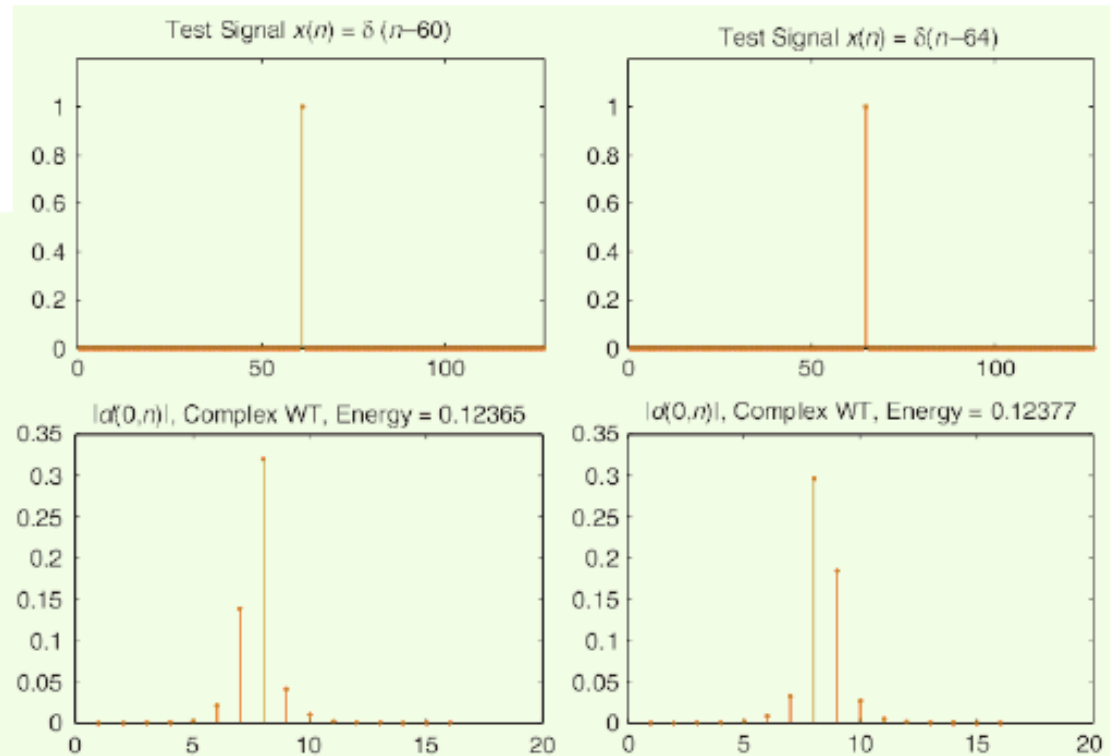
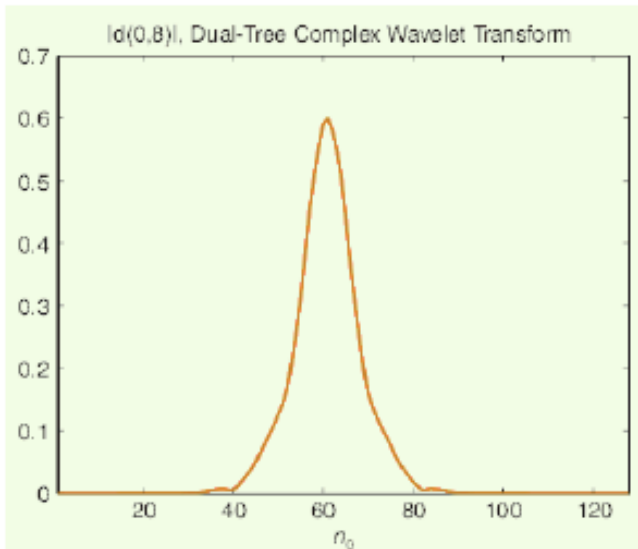
- Use of orthogonal wavelet basis: no redundancy
- Problem: real orthogonal wavelets oscillate too much around discontinuities and are not translation invariant
- Bad for Burgers: we have shocks, and they translate



Complex biorthogonal wavelets

- Complex biorthogonal wavelet basis
- Approximately solves the problem in exchange of a double redundancy

N. Kingsbury, 2001
Appl. Comp. Harm. Anal., **10**, 234

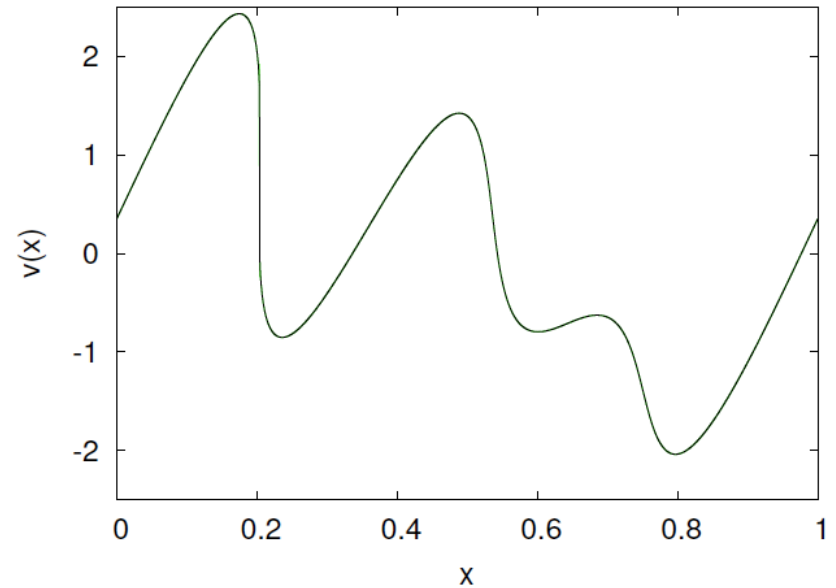
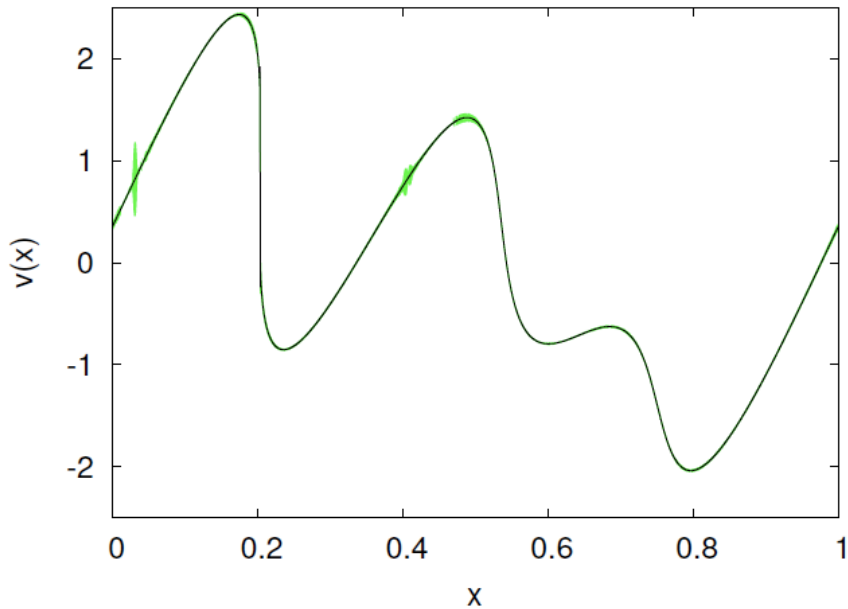


Regularization using complex wavelets

Galerkin-truncated
inviscid solution

Complex wavelet filtered
inviscid solution

Exact solution (black lines)

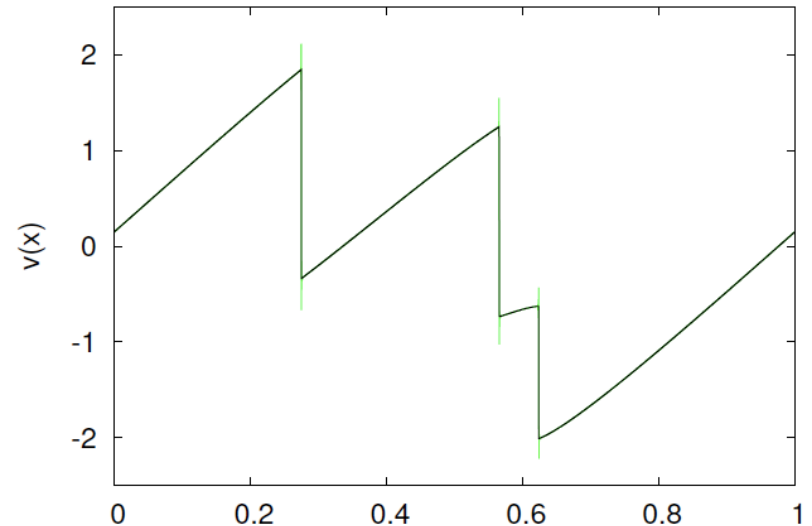
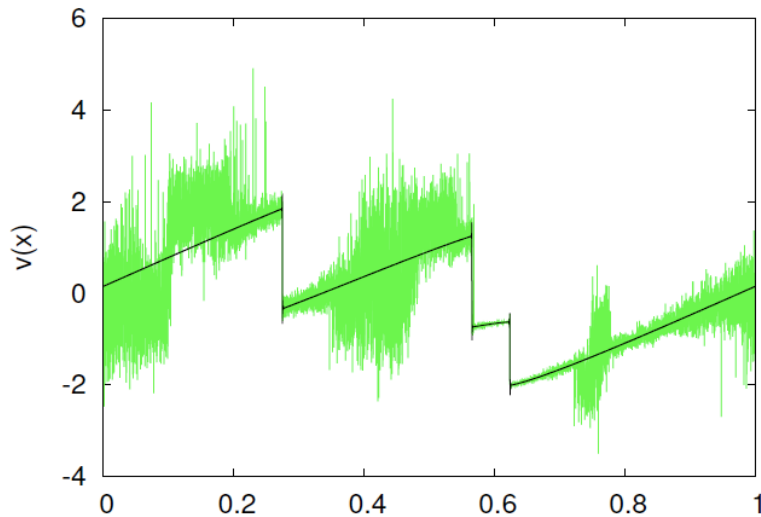
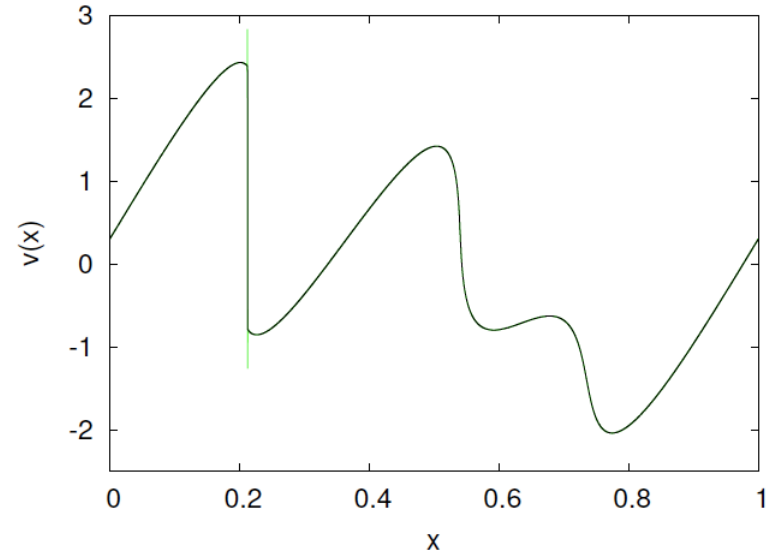
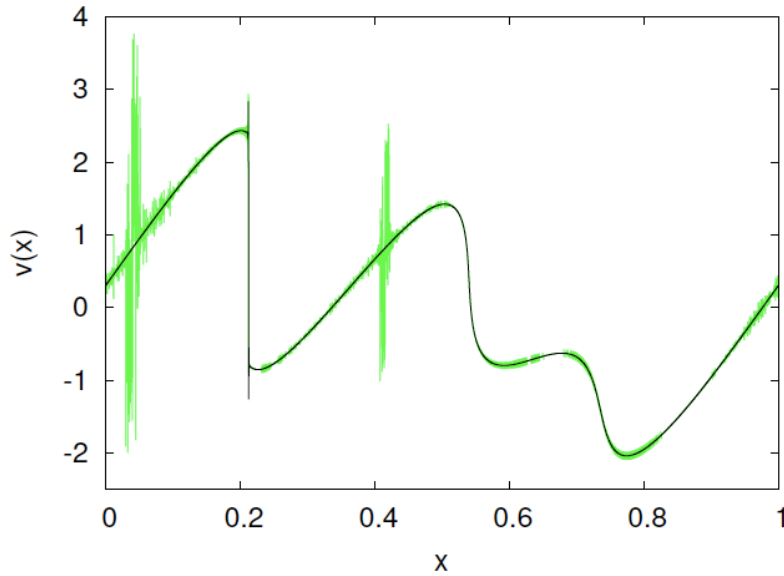


- CVS solution mimics very well the reference solution
- Resonances are completely filtered

Galerkin-truncated inviscid solution

Complex wavelet filtered inviscid solution

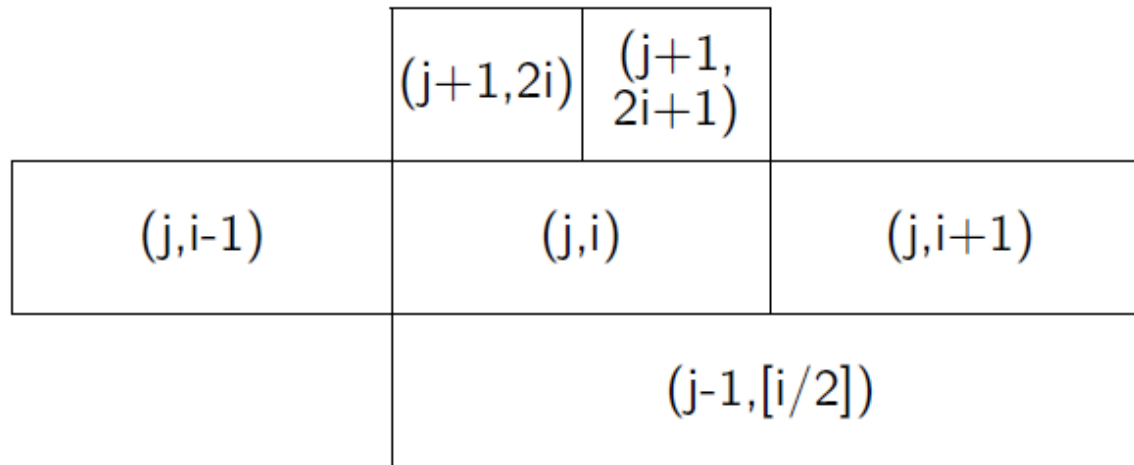
Exact solution (black line)



Addition of a safety zone

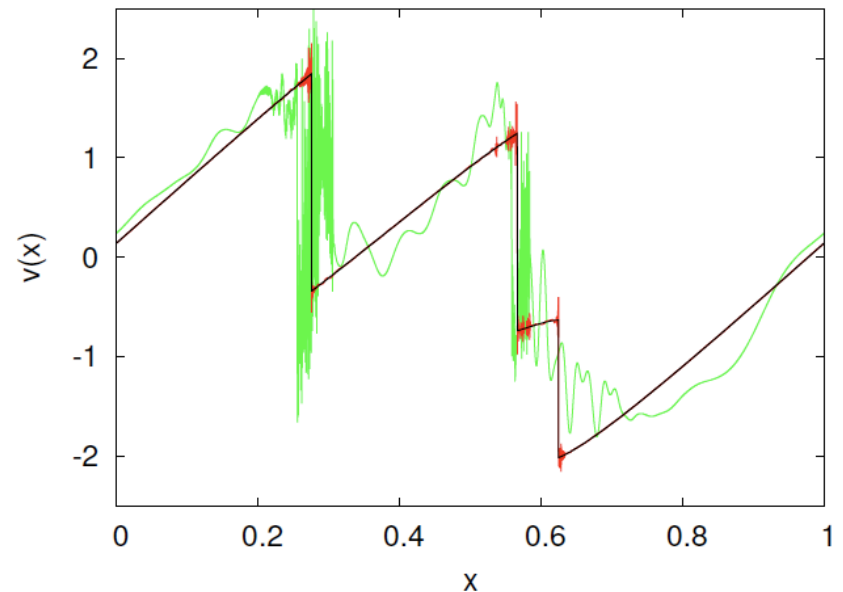
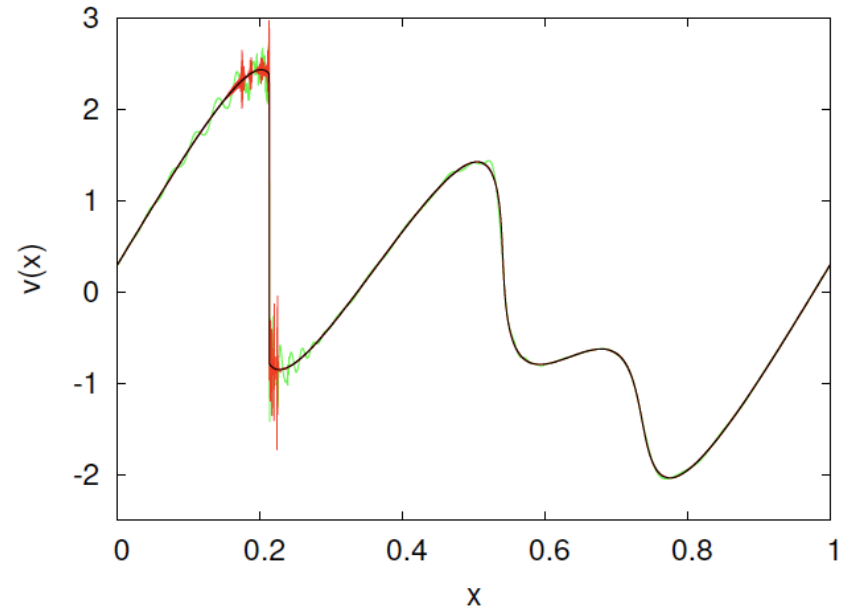
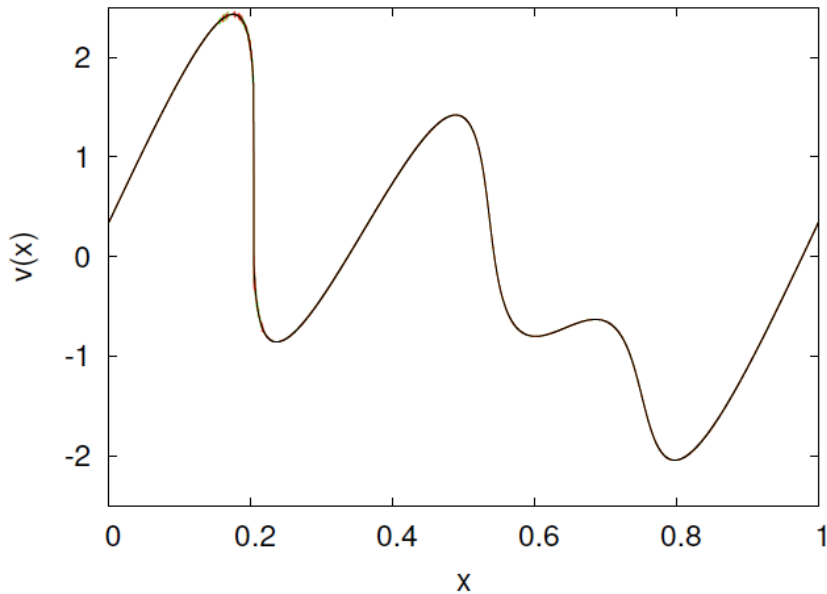
(for real valued wavelets)

- But how to use a real wavelet basis with CVS and Burgers?
- Keep incoherent coefficients neighbor in space and scale to the coherent ones



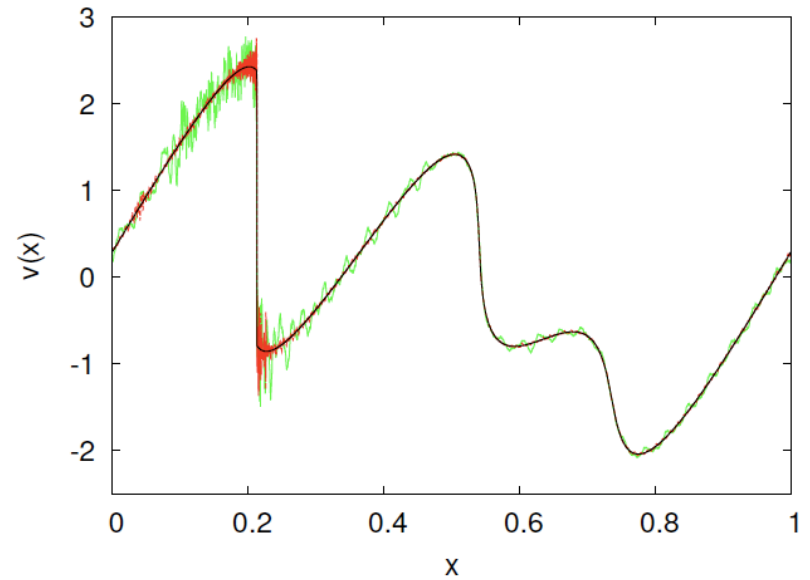
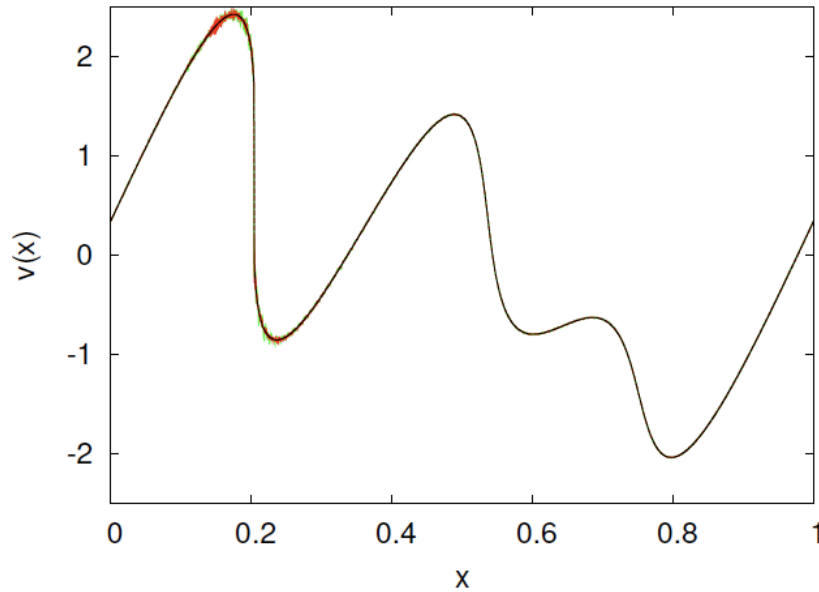
- Try to keep track of translations and generation of finer scales

With or without a safety zone

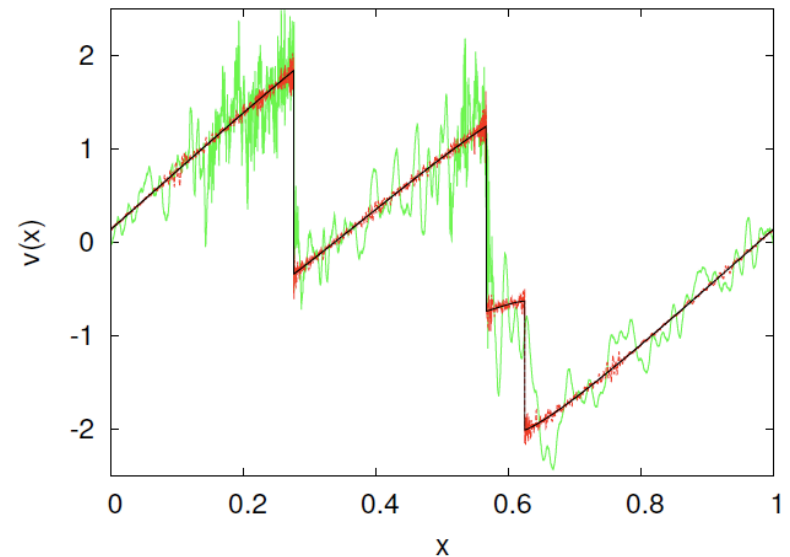


Exact solution
No safety zone added
Safety zone added

Regularization using real wavelets

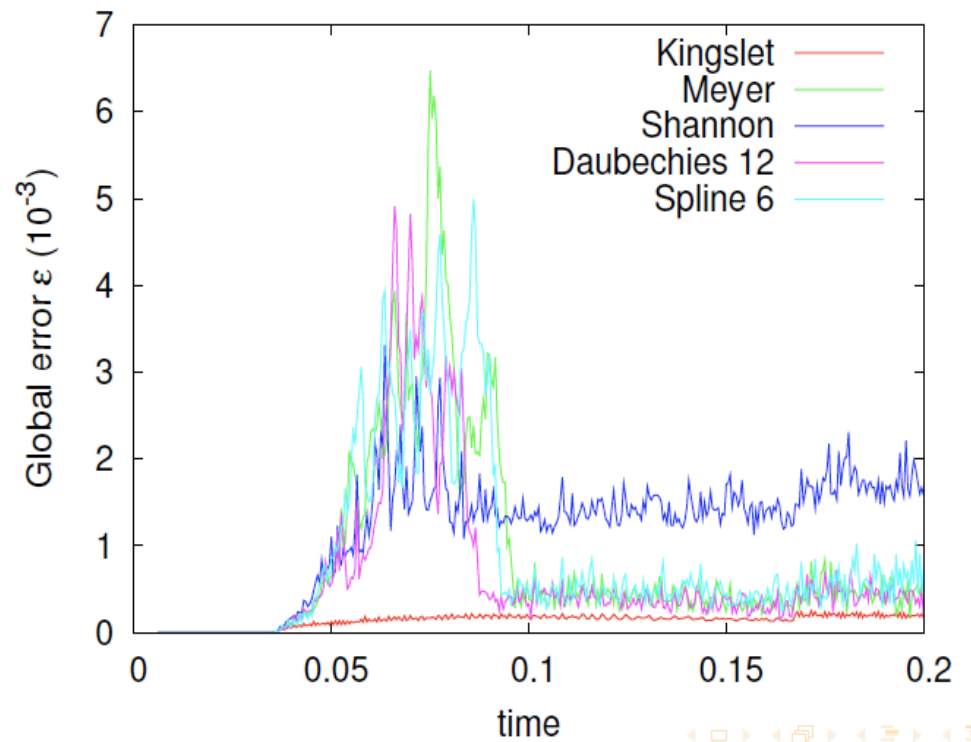
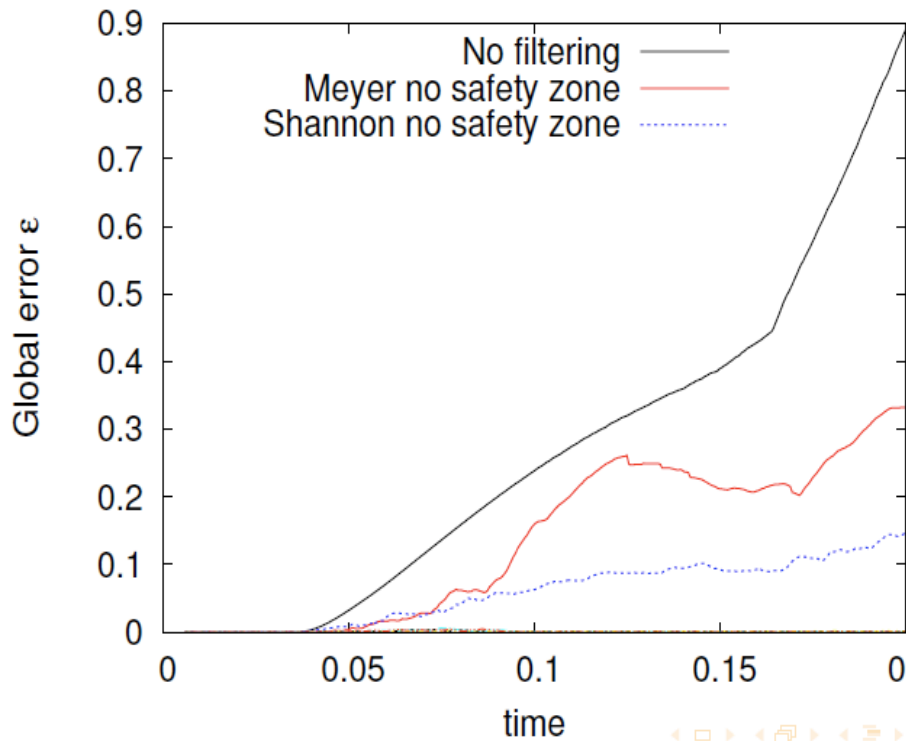


*A safety zone
has been added*

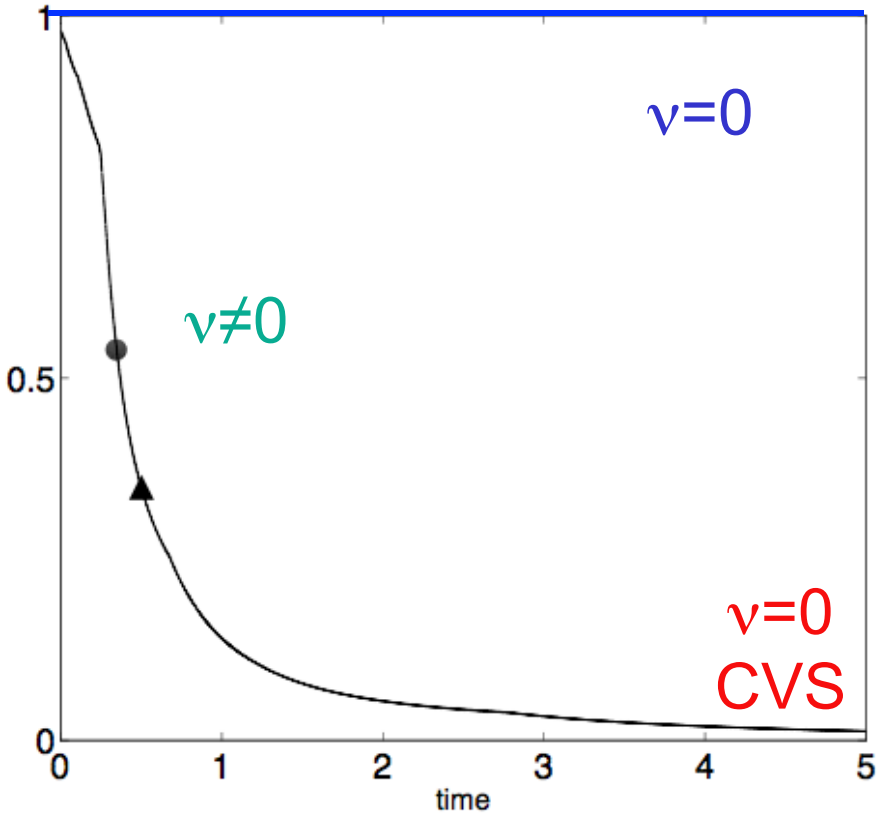


Global error estimation

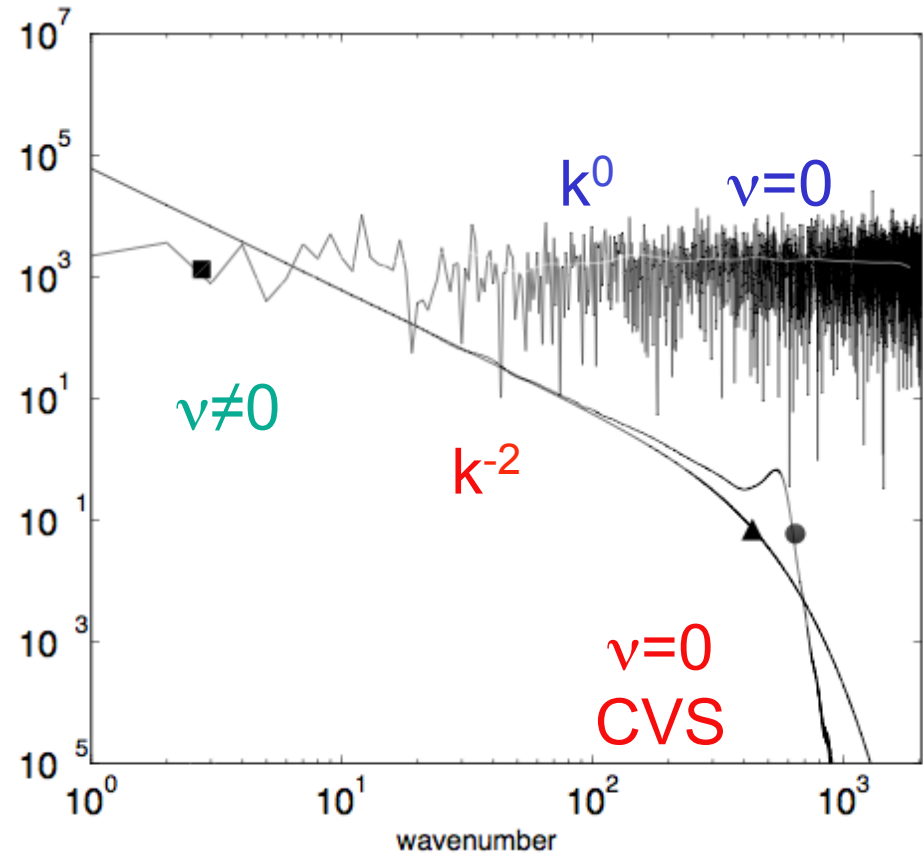
$$\varepsilon = \frac{\int_0^1 [v(x) - v_{\text{ref}}(x)]^2 dx}{\int_0^1 v_{\text{ref}}(x)^2 dx}$$



Deterministic (sine wave) initial condition



Time evolution of energy



Energy spectrum

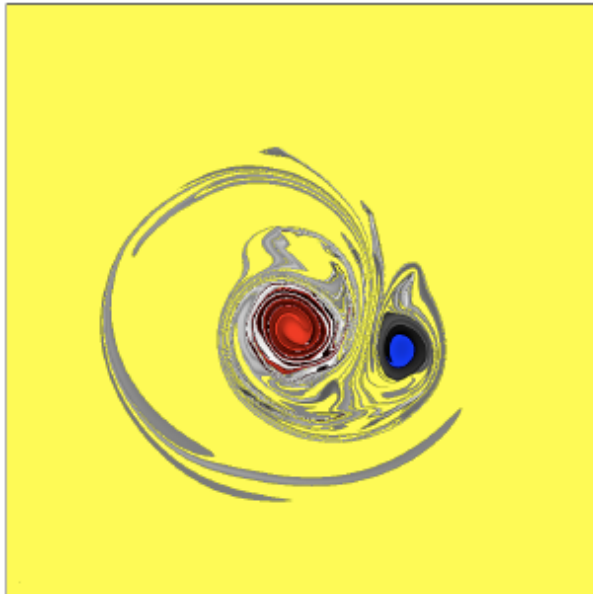
Nguyen, Farge, Kolomenskiy,
Schneider & Kingsbury,
Physica D, 237, 2008

1. Comparison between
viscous, inviscid and wavelet-regularized
Galerkin-truncated 1D Burgers

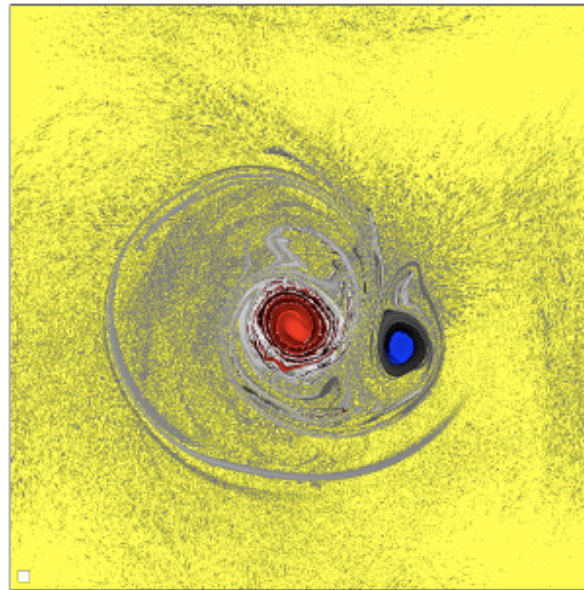
2. Comparison between
2D Navier-Stokes, 2D Euler
and wavelet-regularized 2D Euler

3. Comparison between
3D Navier-Stokes, 3D Euler
and wavelet-regularized 3D Euler

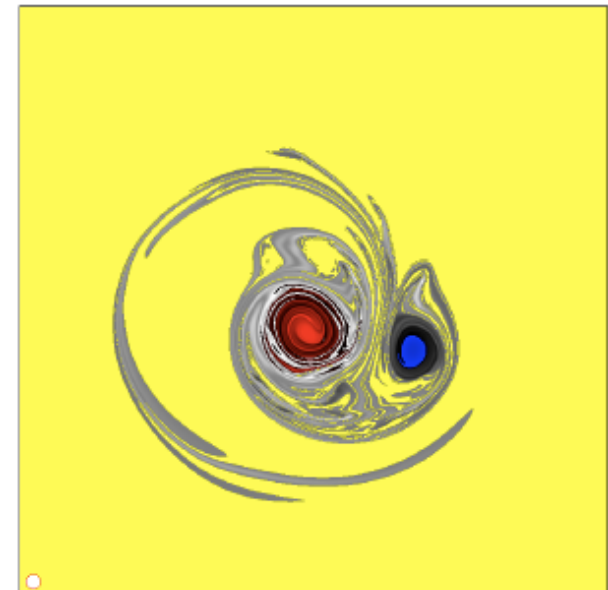
2D three vortex nonlinear interaction



Navier-Stokes
 $\nu > 0$



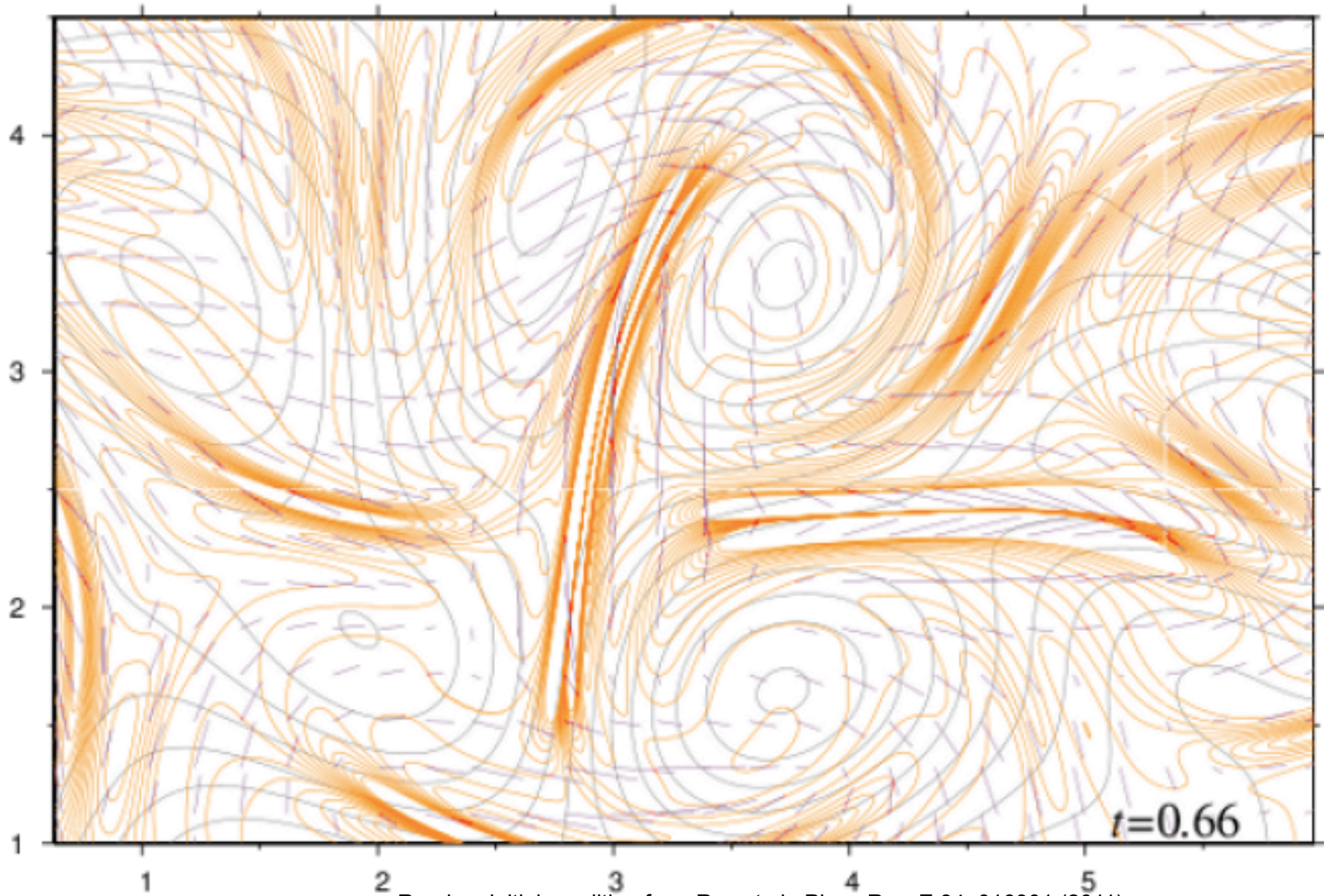
Truncated Euler
 $\nu = 0$



CVS regularized
truncated Euler
 $\nu = 0$

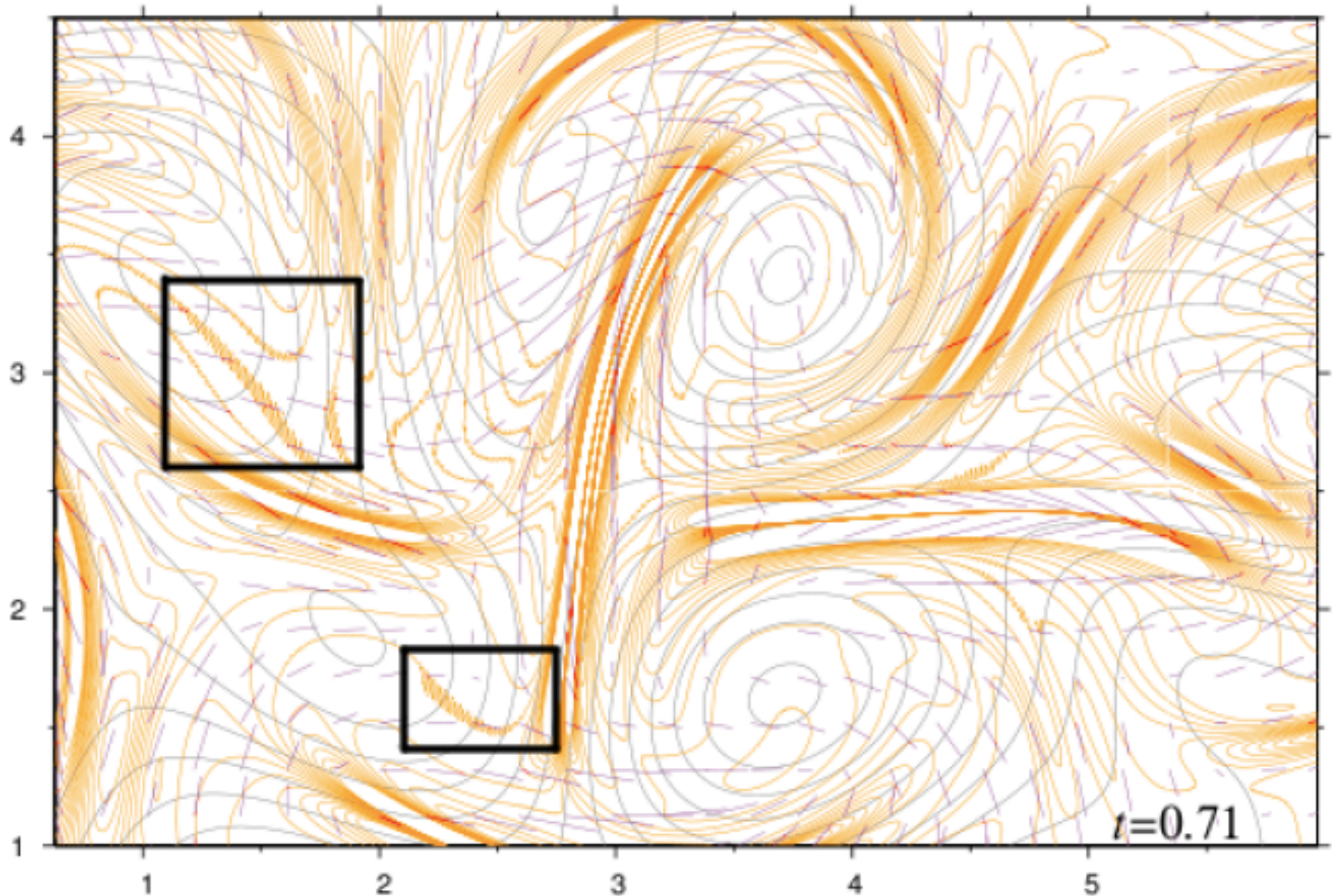
Galerkin-truncated 2D Euler

Laplacian of vorticity and streamlines (grey)



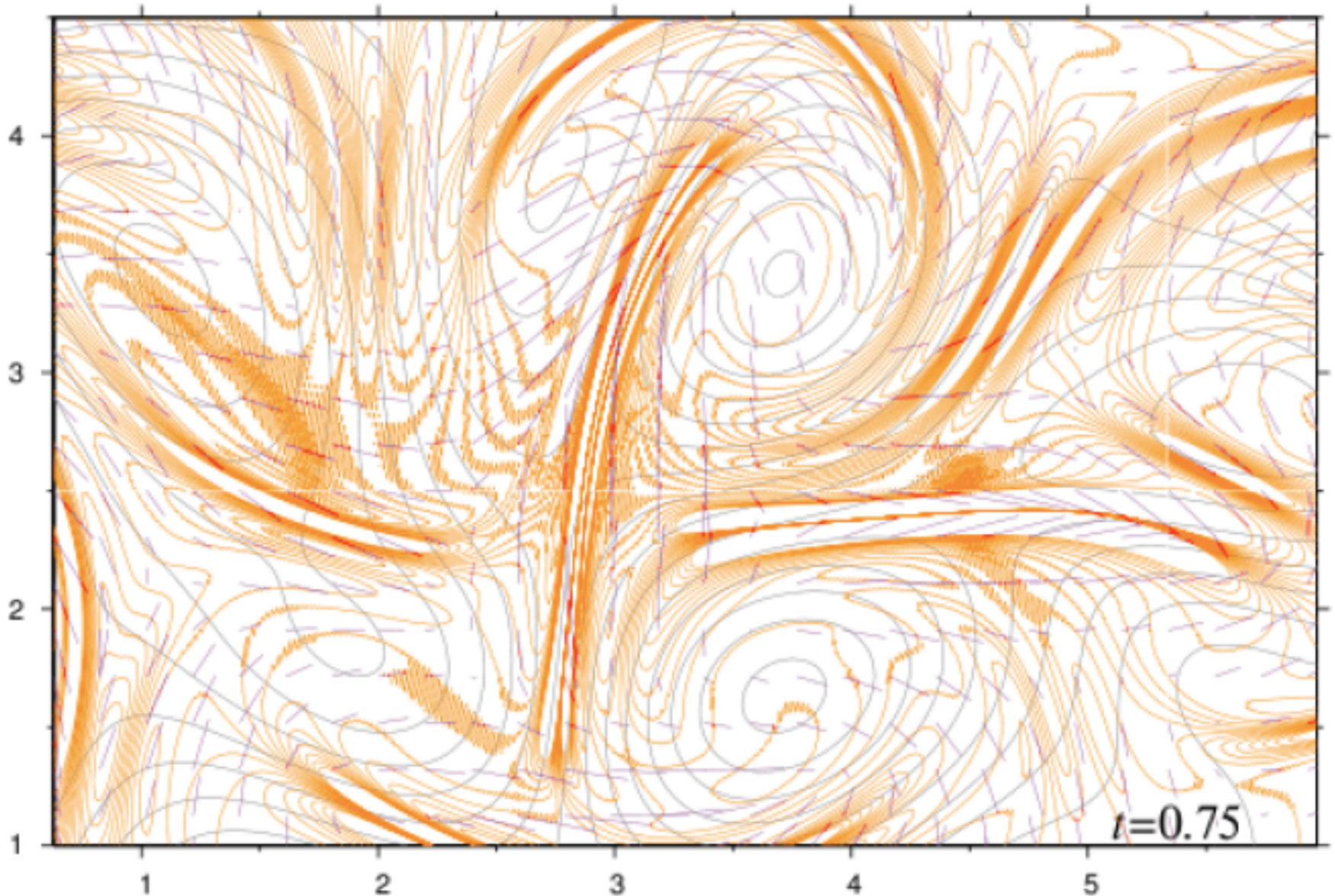
Galerkin-truncated 2D Euler

Laplacian of vorticity and streamlines (grey)



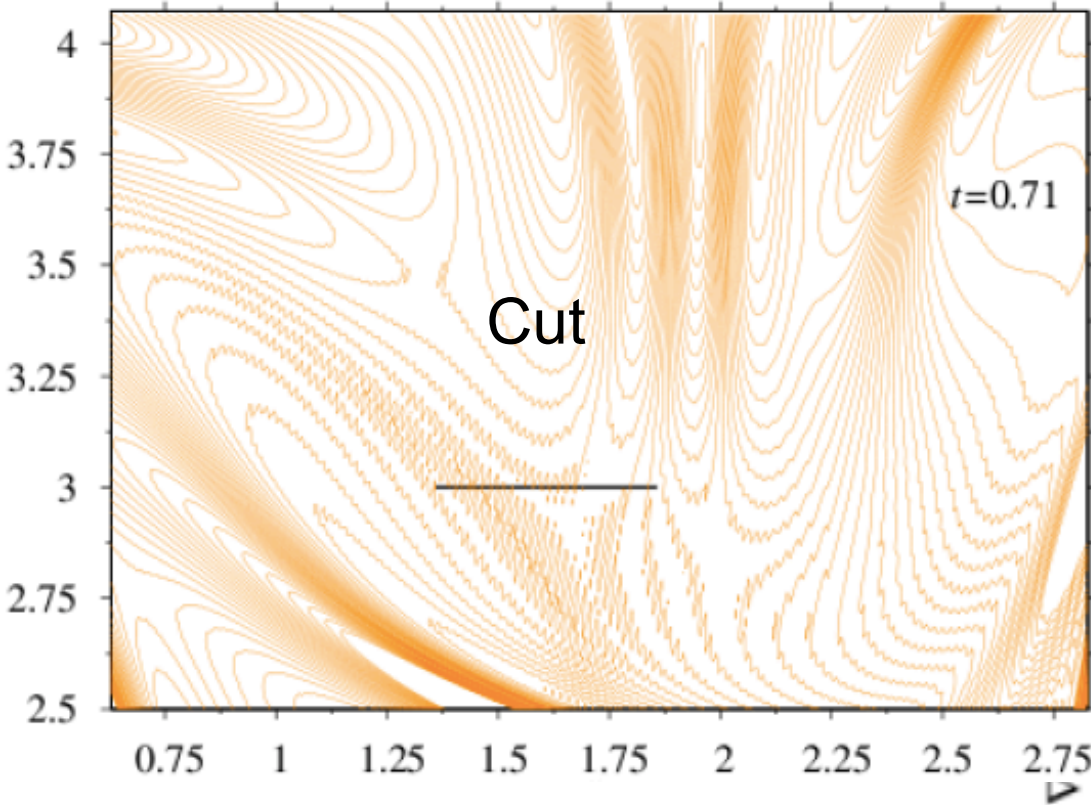
Galerkin-truncated 2D Euler

Laplacian of vorticity and streamlines (grey)



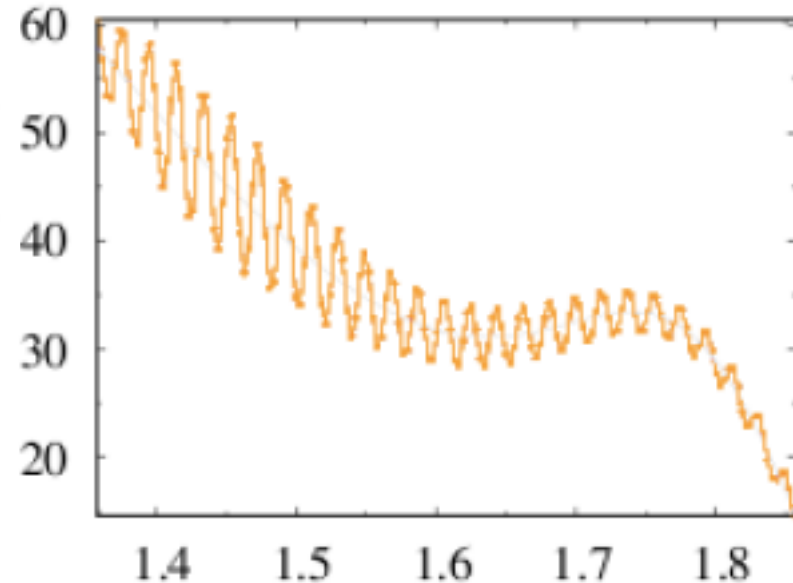
Galerkin-truncated 2D Euler

- Resonances also present in 2D Euler equation



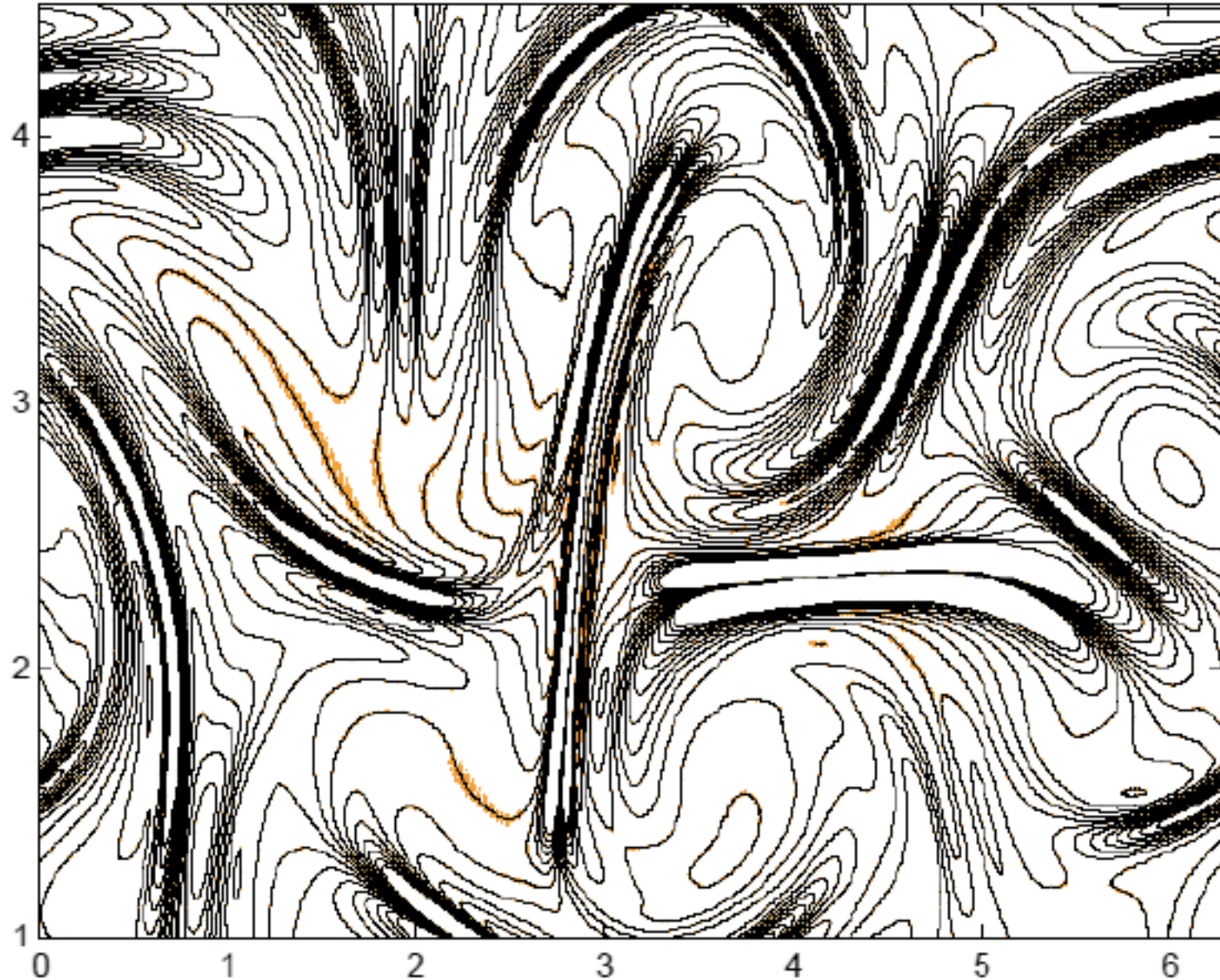
- Contours of $\nabla^2\omega$
Laplacian of vorticity

Cut and zoom

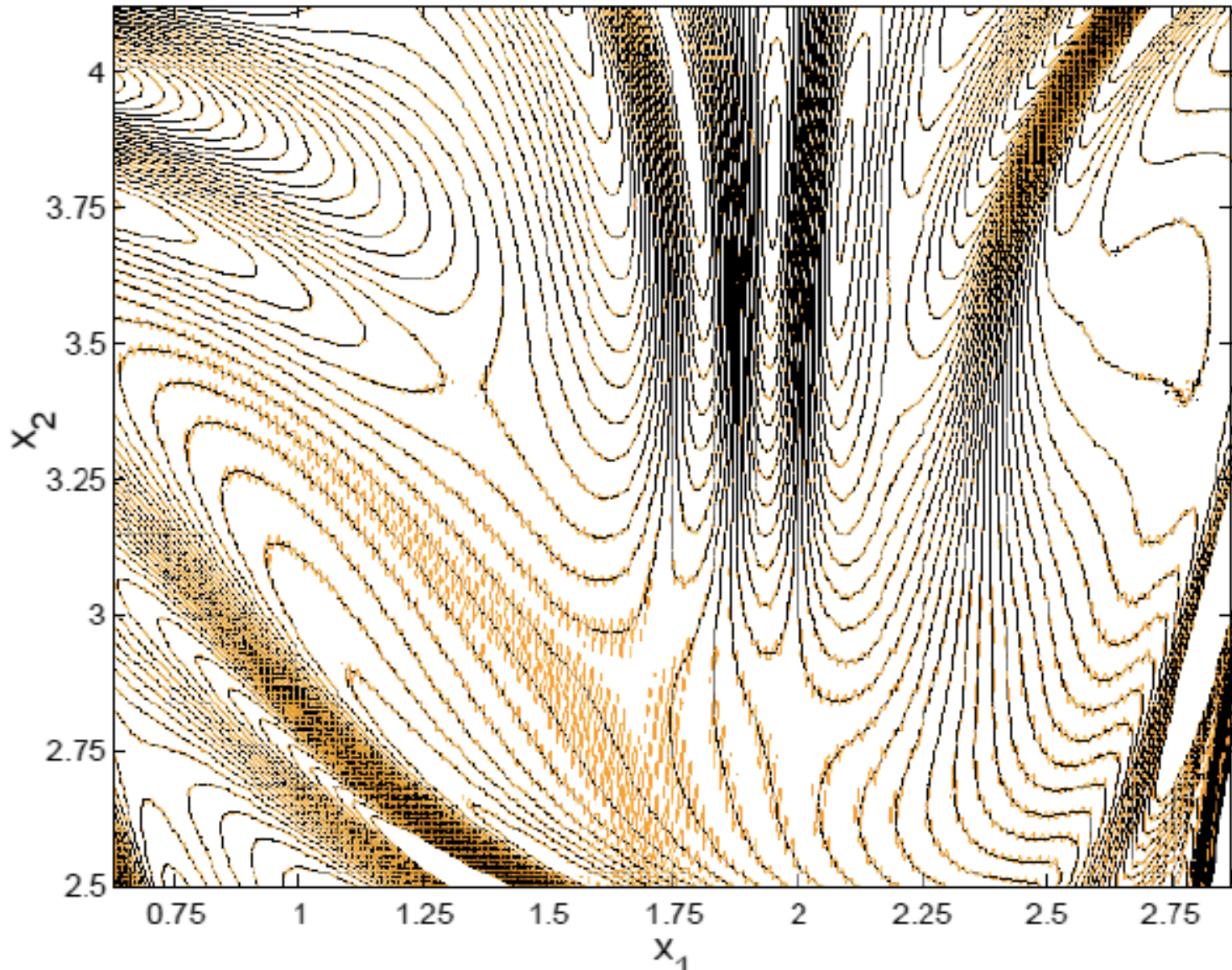


Pereira, Nguyen, Farge,
Schneider, 2013
Phys. Rev. E, 87, 033017

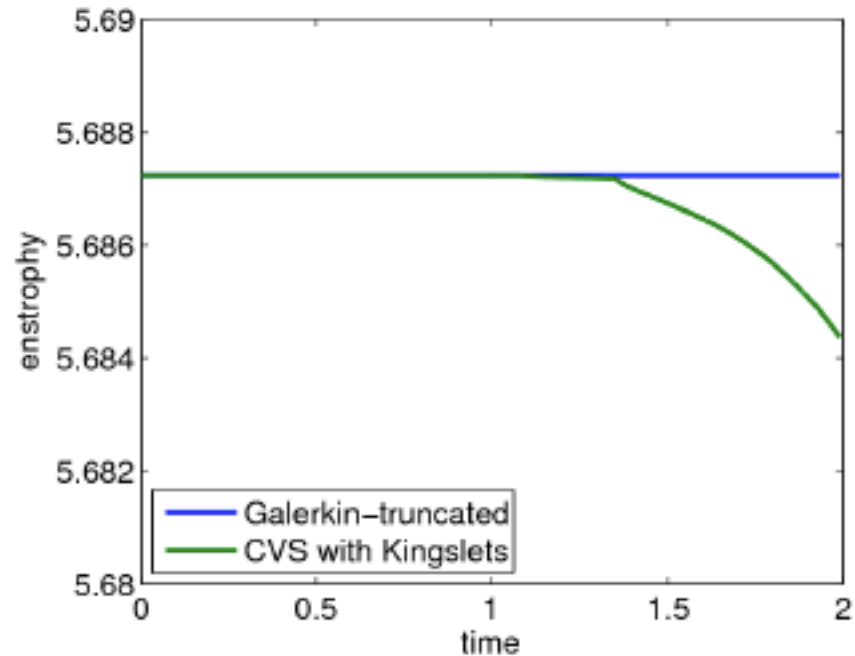
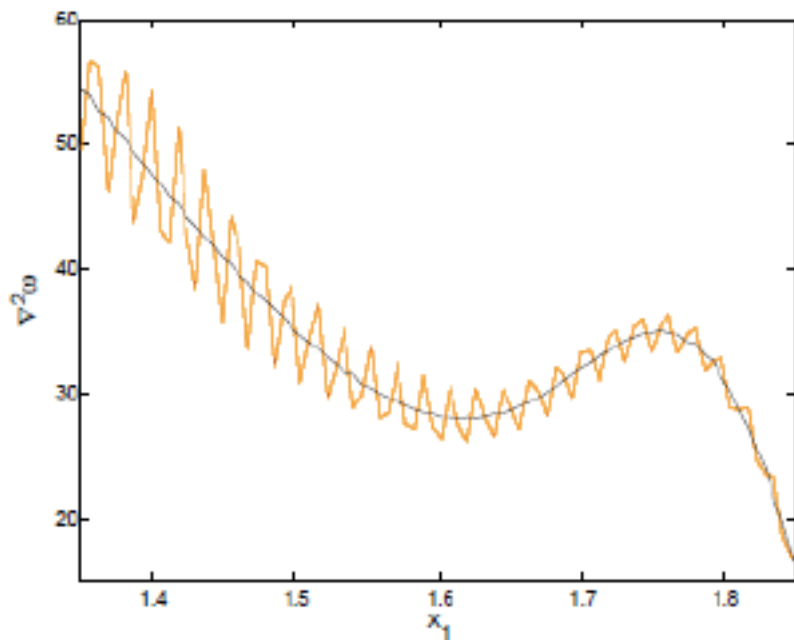
Wavelet filtered / Galerkin-truncated 2D Euler



Wavelet filtered / Galerkin-truncated 2D Euler



Wavelet filtered / Galerkin-truncated 2D Euler



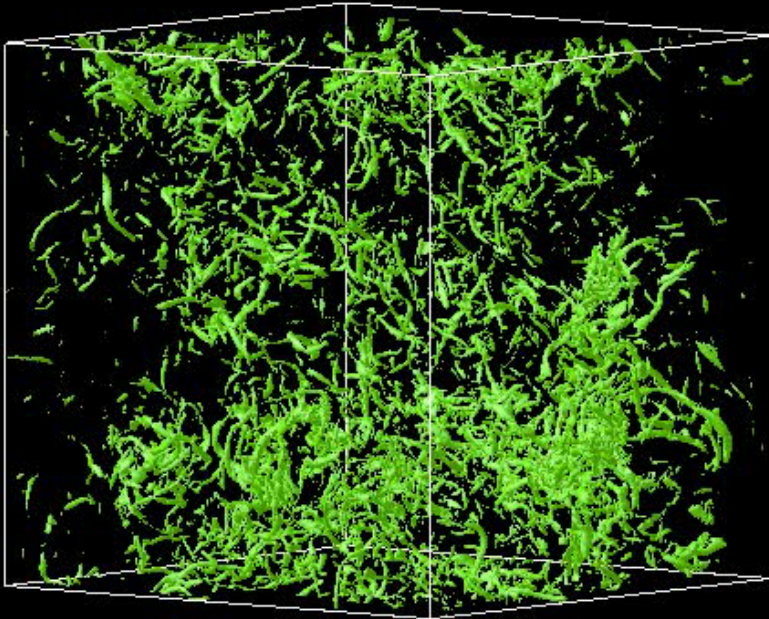
- CVS with Kingslets is able to filter the resonances and keep track of the dynamics, even though the system is chaotic
- CVS is able to discard the unphysical perturbations caused by Galerkin truncation while keeping the dynamics intact
- Real orthogonal wavelets can be used as long as we define a safety zone in wavelet space

1. Comparison between
viscous, inviscid and wavelet-regularized
Galerkin-truncated 1D Burgers

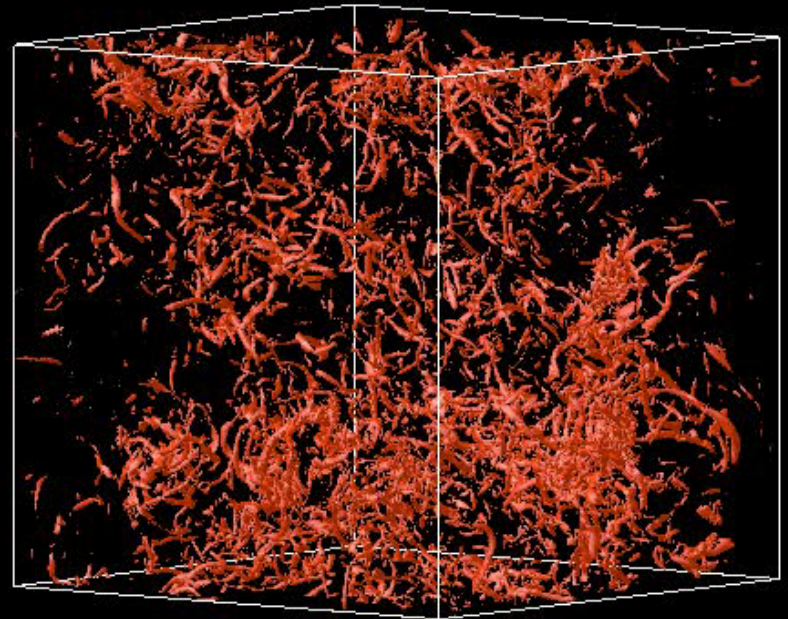
2. Comparison between
2D Navier-Stokes, 2D Euler
and wavelet-regularized 2D Euler

3. Comparison between
3D Navier-Stokes, 3D Euler
and wavelet-regularized 3D Euler

3D decaying Navier-Stokes turbulence

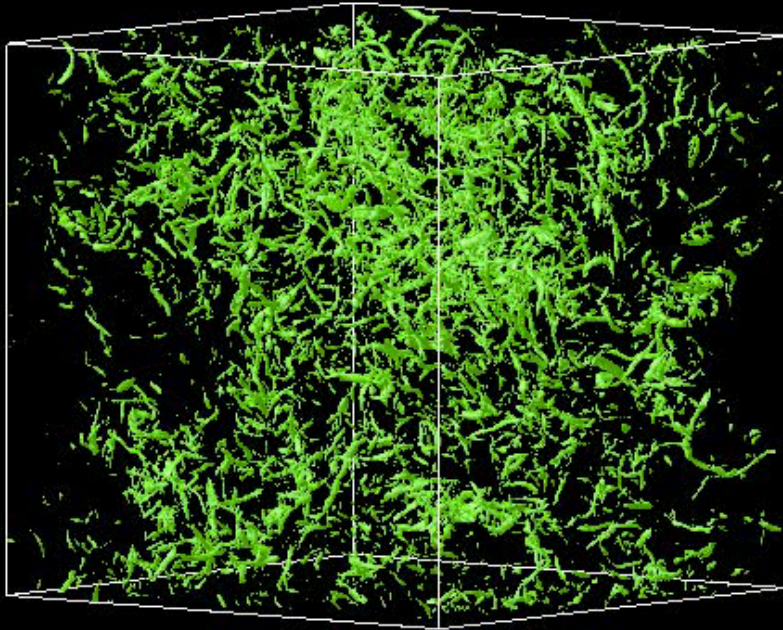


DNS
 $N=256^3$

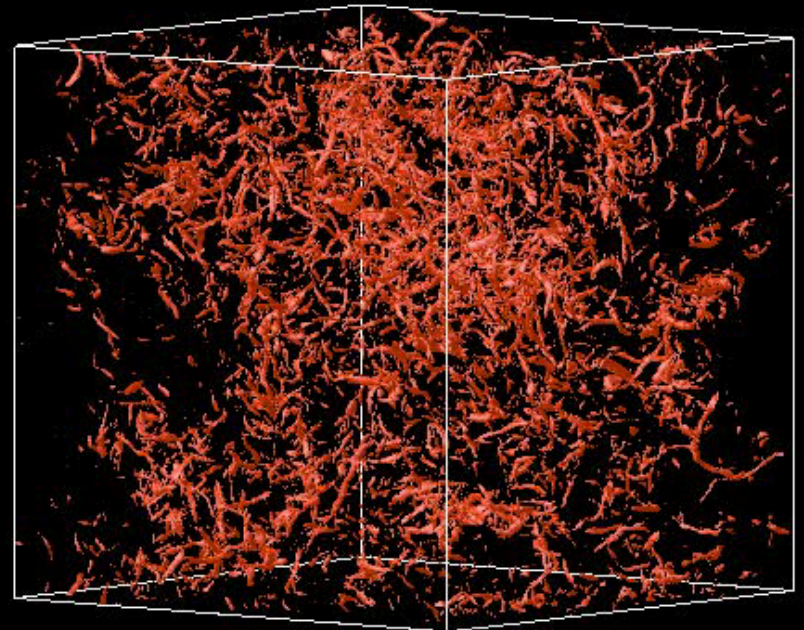


CVS (wavelet filtered)
 $N_c=(2\%+10\%)N$

3D forced Navier-Stokes turbulence

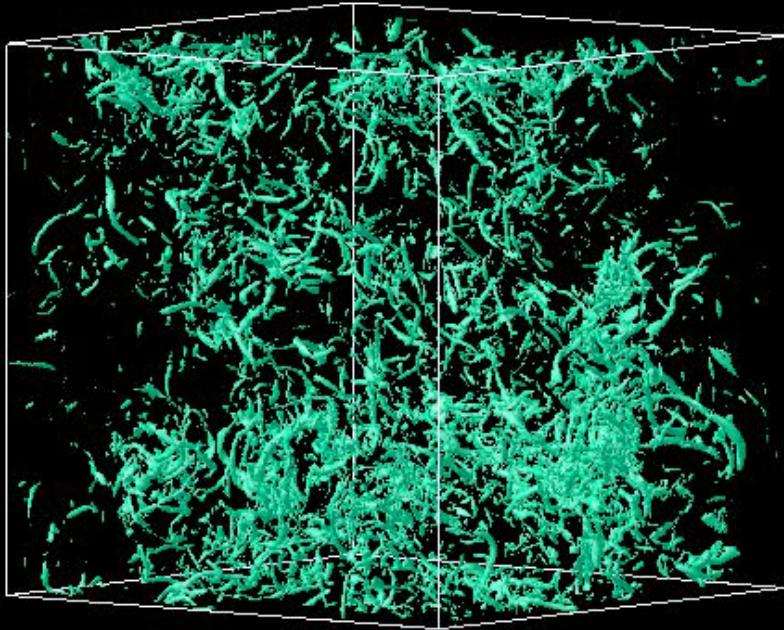


DNS
 $N=256^3$

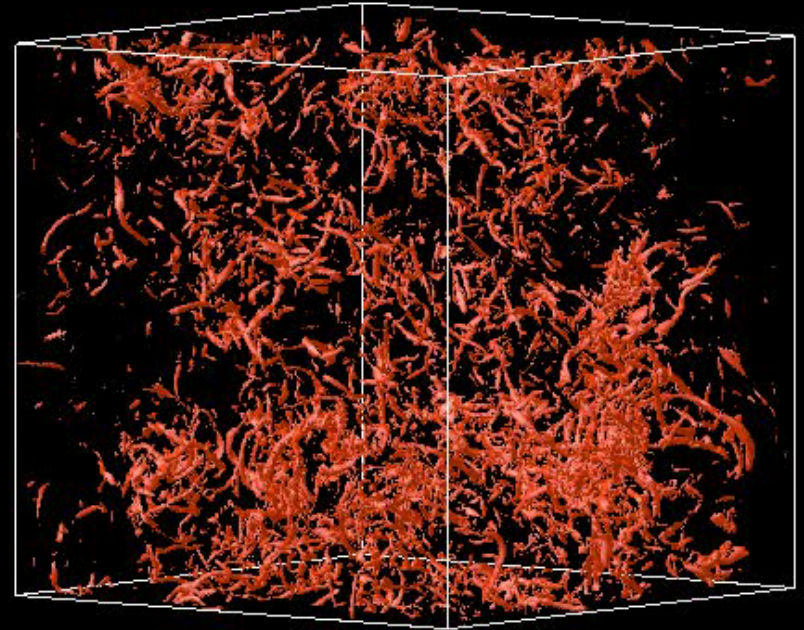


CVS (wavelet filtered)
 $N_c=(2\%+10\%)N$

3D Euler turbulence

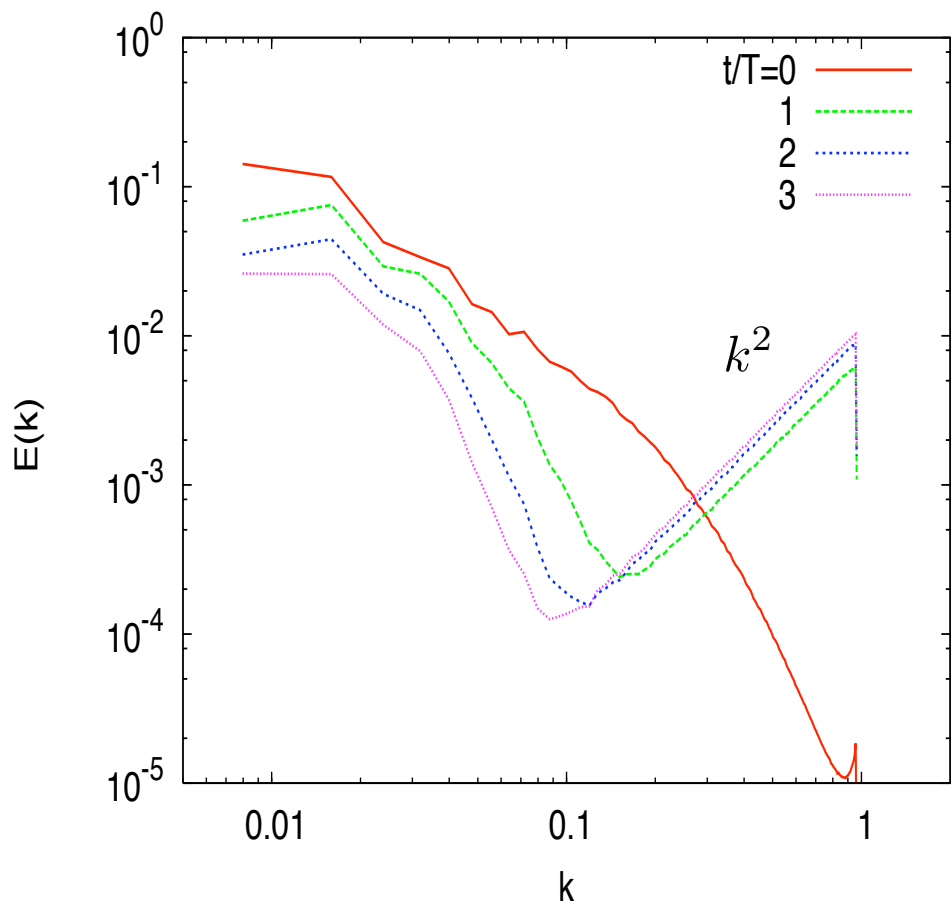


DNS
 $N=256^3$

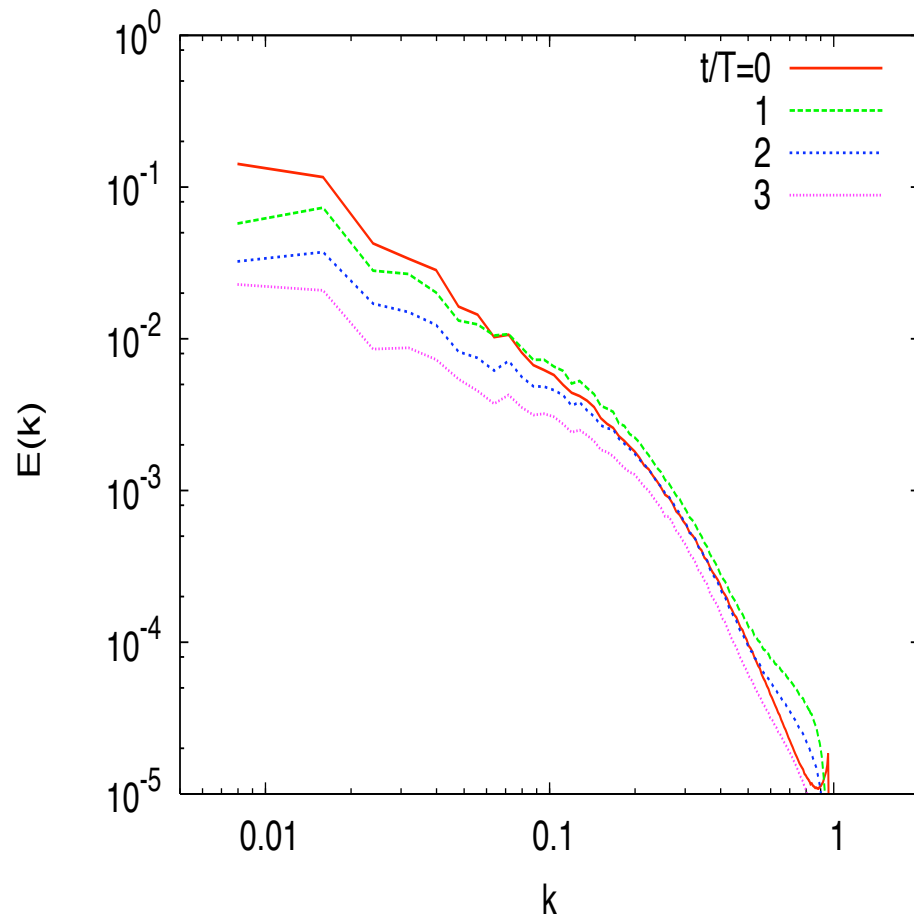


CVS (wavelet filtered)
 $N_c=(2\%+10\%)N$

CVS of 3D Euler turbulence



DNS
 $N=256^3$



CVS
 $N_c=(2\%+10\%)N$

Conjecture/conclusions

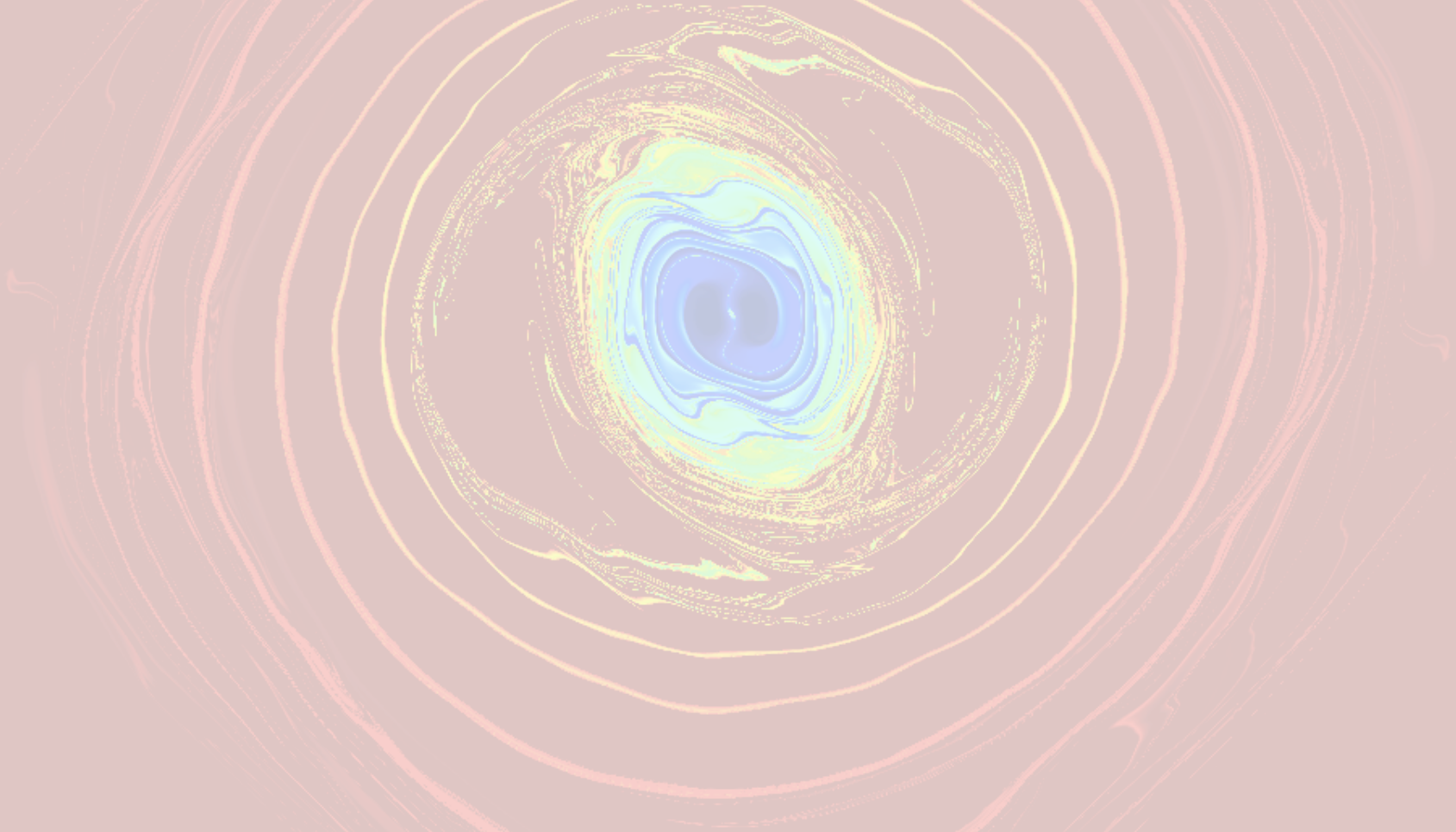
Wavelet regularizing the truncated **inviscid 1D Burgers** ($\nu=0$) is equivalent to computing the **continuous viscous 1D Burgers** ($\nu>0$) while preserving the **dissipation due to singularities**.

Wavelet regularizing the truncated incompressible **2D Euler** ($\nu=0$) is equivalent to computing the incompressible **2D Navier-Stokes** ($\nu>0$).

Wavelet regularizing the truncated incompressible **3D Euler** ($\nu=0$) might be equivalent to computing the incompressible **3D Navier-stokes** ($\nu>0$) while preserving the **dissipation due to possible singularities**.

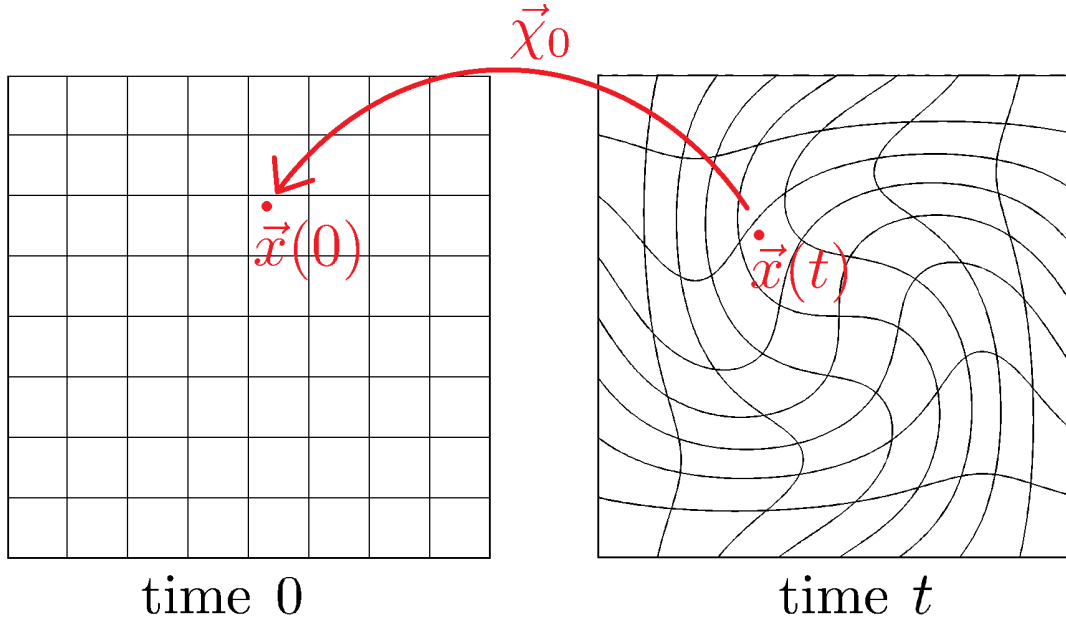
A CHARACTERISTIC MAPPING METHOD FOR 2D EULER

A SEMI LAGRANGIAN METHOD TO SOLVE THE 2D INCOMPRESSIBLE EULER EQUATIONS



Assume we have a diffeomorphism $\vec{\chi}_0(\vec{x}, t) : \mathbb{R}^d \supset U \rightarrow U$ such that for any initial point $\vec{x}(t=0) \in U$ being evolved under the vector field \vec{v} we have

$$\vec{\chi}_0(\vec{x}(t), t) = \vec{x}(0).$$



we can decompose the diffeomorphism into n pieces:

$$\vec{\chi}_0(\vec{x}, T) = \vec{\chi}_1(\vec{\chi}_2(\dots \vec{\chi}_n(\vec{x}, \tau_n) \dots, \tau_2), \tau_1)$$

each piece solves the following problem:

$$\frac{\partial \vec{\chi}_i}{\partial t} + \vec{v} \cdot \nabla \vec{\chi}_i = 0 \quad \forall t \geq \tau_{i-1}, x \in U$$

$$\vec{\chi}_i(\vec{x}, \tau_{i-1}) = \vec{x}.$$

2D incompressible Euler Equations

$$\omega \equiv \nabla \times \vec{u}$$

$$\omega_t + \vec{u} \cdot \nabla \omega = 0$$

$$\vec{u} = \nabla^\perp \Delta^{-1} \omega$$

- Evolve the diffeomorphism using the CM approach (Hermite jet scheme, gradient augmented level set)
- Compute the vorticity on the fine grid
- Compute the velocity using FFT \rightarrow filter on the CM grid

$$\omega(\vec{x}, t) = \omega_0(\chi(\vec{x}, t))$$

Remapping criterion $|\det(\nabla \chi_i) - 1| < \epsilon$

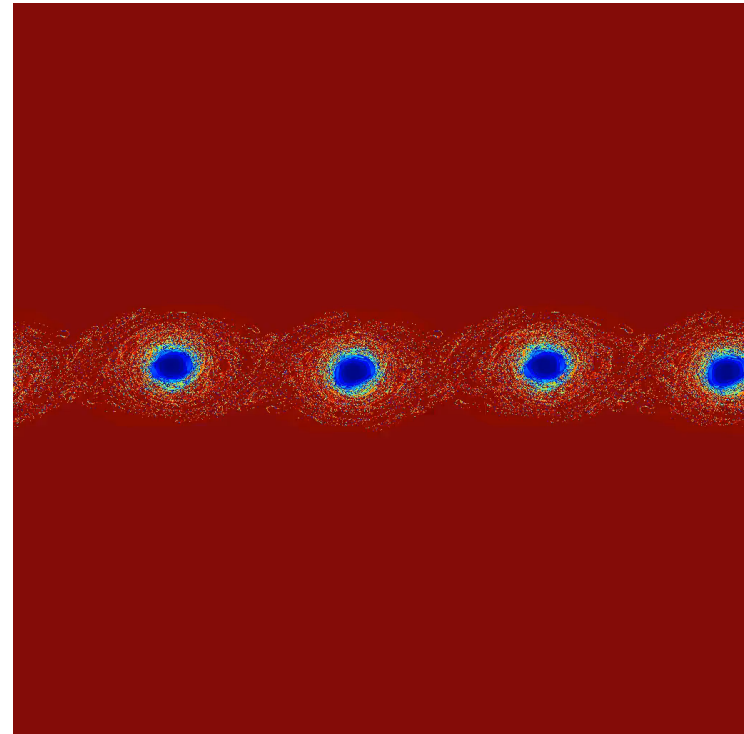
- The velocity acts on all scales of vorticity
- The large scale vorticity contributes to the velocity

A CHARACTERISTIC MAPPING METHOD FOR 2D EULER

EXAMPLE: KELVIN HELMHOLTZ INSTABILITY



insect_rigid_turb.avi



insect_rigid_turb.avi

Numerics: A characteristic mapping method with exponential resolution.

In collaboration with B. Yadav and J.C. Nave, *Bull. Amer. Phys. Soc*, 61(20), 2016.

Summary

- Gradient-Augmented Level Set Method (GALSM)
- Arbitrary sets using jet schemes
- 2D incompressible Euler (preprint available soon)
- Current/future work: Extension to 3D / generalize the framework

