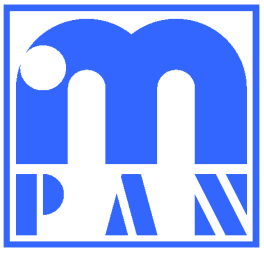
INSTITUTE OF MATHEMATICS
of the
Polish Academy of Sciences



ul. Śniadeckich 8, P.O.B. 21, 00-956 Warszawa 10, Poland http://www.impan.gov.pl

IM PAN Preprint 684 (2007)

Erwin Deriaz, Valérie Perrier

Direct Numerical Simulation of turbulence using divergence-free wavelets

Presented by Teresa Regińska

Published as manuscript

Received 29 June 2007

Direct Numerical Simulation of turbulence using divergence-free wavelets

Erwan Deriaz * Valérie Perrier †

June 29, 2007

Abstract

We give a numerical method based on divergence-free wavelets to solve the incompressible Navier-Stokes equations. We present a new scheme which uses anisotropic (or generalized) divergence-free wavelets, and which only needs Fast Wavelet Transform algorithm. We prove its stability and show convincing numerical experiments.

Introduction

A high degree of complexity arises in the numerical solution of the incompressible Navier-Stokes equations due to the creation and interaction of small scales in the turbulent flows. Existing numerical methods applied to the computation of approximate solutions of Navier-Stokes equations can be distributed into two main groups: on one hand, the ones well-localized in space (finite differences, finite elements and finite volumes) and on the other hand, the ones well-localized in frequency (spectral methods). Wavelet methods offer the possibility to mix both and to graduate the accuracy in space and in frequency. However, a function ψ (for instance a wavelet) cannot be completely accurate both in space and in frequency: the Heisenberg inequality states that $\Delta x \Delta\xi \geq 1/2$ for Δx the variance of ψ, and $\Delta\xi$ the variance of $\widehat{\psi}$.

At the beginning, reaserchers in fluid mechanics used wavelets to analyse turbulent flows [28, 12, 18]. Then wavelet codes for the numerical simulation of Naver-Stokes equations made an apparition [2, 20]. They looked promising, especially concerning adaptivity issues [15]. Among wavelet methods, we can notice adaptive collocation methods [23], and coherent vortex extraction [14] which makes use of the wavelet/vaguelette decomposition and relies on denoising for adaptivity. Other authors [17] applied wavelet/Galerkin methods to the Navier-Stokes equations.

Here, we will use divergence-free wavelets as an algorithmic solver of PDE's and for adaptive purpose. We get a maximum benefit from the localization in space allowed by wavelets, provided that the accuracy in frequency still permits competitive algorithms. For this purpose, we'll use anisotropic divergence-free and curl-free wavelets introduced

*Institute of Mathematics, Polish Academy of Sciences. ul. Sniadeckich 8, 00-956 Warszawa, Poland, to whom correspondance should be addressed (E.Deriaz@impan.gov.pl)

†Laboratoire Jean Kuntzmann, INPG, BP 53, 38 041 Grenoble cedex 9, France (Valerie.Perrier@imag.fr)

in [9]. They naturally arise with an iterative Leray projector (deriving from a Helmholtz decomposition) mimicking the spectral case and whose convergence is proved in [10].

We also use a wavelet solver for the implicit Laplacian to construct a wavelet semi-implicit numerical scheme for the simulation of incompressible Navier-Stokes equations.

Hence, the numerical solution of the Navier Stokes equations we present, requires *only wavelet transforms* and no other computational tool such as Fourier Transform or linear system solvers (Conjugate Gradient, GMRES...). This permits using it in a completely adaptive context. On top of this, wavelet decomposition comes naturally with a scale partition which should enable adaptive treatment of computations.

The scheme we propose is proved to be stable under a Courant-Friedrich-Levy (CFL) condition, provided a sufficiently smooth solution to the Naver-Stokes equations exists. The stability of this wavelet scheme is mainly due to the divergence-free condition which is automatically and exactly satisfied by the divergence-free wavelet decomposition of the solution.

First, we'll briefly introduce the divergence-free and curl-free wavelets, and present the wavelet Helmholtz decomposition. Then we'll present the Navier-Stokes scheme and its numerical analysis, followed by a numerical experiment with the simulation of the merging of three vortices. Finally, we'll give an insight on how to make this method adaptive, with the support of some experiments.

1 Divergence-free and curl-free wavelets

The wavelet Helmholtz decomposition arises from divergence-free and gradient (i.e. curl-free) wavelet decompositions. These wavelets are constructed thanks to two 1-D wavelets ψ_0 and ψ_1 related by differentiation: $\psi_1'(x) = 4\ \psi_0(x)$ [24].

1.1 Divergence-free wavelets

The 2D anisotropic divergence-free wavelets are generated from a single vector function

$$\Psi^{\mathrm{div}}(x_1, x_2) = \left| \begin{array}{l} \psi_1(x_1)\psi_0(x_2) \\ -\psi_0(x_1)\psi_1(x_2) \end{array} \right.$$

by anisotropic dilations, and translations. The 2D anisotropic divergence-free wavelets are given by:

$$\Psi^{\mathrm{div}}_{\mathbf{j},\mathbf{k}}(x_1, x_2) = \left| \begin{array}{l} 2^{j_2}\psi_1(2^{j_1}x_1 - k_1)\psi_0(2^{j_2}x_2 - k_2) \\ -2^{j_1}\psi_0(2^{j_1}x_1 - k_1)\psi_1(2^{j_2}x_2 - k_2) \end{array} \right.$$

where $\mathbf{j} = (j_1, j_2) \in \mathbb{Z}^2$ is the scale parameter, and $\mathbf{k} = (k_1, k_2) \in \mathbb{Z}^2$ is the position parameter. For $\mathbf{k}, \mathbf{j} \in \mathbb{Z}^2$, the family $\{\Psi^{\mathrm{div}}_{\mathbf{j},\mathbf{k}}\}$ forms a basis of $\mathbf{H}_{\mathrm{div},0}(\mathbb{R}^2) = \{\mathbf{f} \in (L^2(\mathbb{R}^2))^2 / \mathrm{div}\ \mathbf{f} \in L^2(\mathbb{R}^2), \quad \mathrm{div}\ \mathbf{f} = 0\}$.
We introduce

$$\Psi^{\mathrm{n}}_{\mathbf{j},\mathbf{k}}(x_1, x_2) = \left| \begin{array}{l} 2^{j_1}\psi_1(2^{j_1}x_1 - k_1)\psi_0(2^{j_2}x_2 - k_2) \\ 2^{j_2}\psi_0(2^{j_1}x_1 - k_1)\psi_1(2^{j_2}x_2 - k_2) \end{array} \right.$$

as complement functions since $\Psi^{\mathrm{n}}_{\mathbf{j},\mathbf{k}}$ is orthogonal to $\Psi^{\mathrm{div}}_{\mathbf{j},\mathbf{k}}$ ($\mathbf{j}, \mathbf{k}$ being fixed).

A similar construction is given for dimension three in [9, 10].

1.2 Curl-free wavelets

The construction of curl-free wavelets (*i.e.* gradient wavelets) is similar to the construction of divergence-free wavelets, despite some crucial differences. The starting point here is to find wavelets in the MRA $(V_j^0 \otimes V_j^1) \times (V_j^1 \otimes V_j^0)$ instead of $(V_j^1 \otimes V_j^0) \times (V_j^0 \otimes V_j^1)$, where the one-dimensional spaces V^0 and V^1 are related by differentiation and integration.

Let $\mathbf{H}_{\mathrm{curl},0}(\mathbb{R}^2)$ be the space of gradient functions in $L^2(\mathbb{R}^2)$. We construct gradient wavelets by taking the gradient of a 2D wavelet basis of the MRA $(V_j^1 \otimes V_j^1)$. If we neglect the L^2-normalization, the anisotropic gradient wavelets are defined by:

$$\Psi^{\mathrm{curl}}_{\mathbf{j},\mathbf{k}}(x_1,x_2) = \frac{1}{4}\,\nabla\left(\psi_1(2^{j_1}x_1-k_1)\psi_1(2^{j_2}x_2-k_2)\right) = \left| \begin{array}{l} 2^{j_1}\psi_0(2^{j_1}x_1-k_1)\psi_1(2^{j_2}x_2-k_2) \\ \\ 2^{j_2}\psi_1(2^{j_1}x_1-k_1)\psi_0(2^{j_2}x_2-k_2) \end{array} \right.$$

Thus, for $\mathbf{j} = (j_1, j_2), \mathbf{k} = (k_1, k_2) \in \mathbb{Z}^2$, the family $\{\Psi^{\mathrm{curl}}_{\mathbf{j},\mathbf{k}}\}$ forms a wavelet basis of $\mathbf{H}_{\mathrm{curl},0}(\mathbb{R}^2)$. We complete this basis to a $\left(L^2(\mathbb{R}^2)\right)^2$-basis with the following complement wavelets:

$$\Psi^{\mathrm{N}}_{\mathbf{j},\mathbf{k}}(x_1,x_2) = \left| \begin{array}{l} 2^{j_2}\psi_0(2^{j_1}x_1-k_1)\psi_1(2^{j_2}x_2-k_2) \\ \\ -2^{j_1}\psi_1(2^{j_1}x_1-k_1)\psi_0(2^{j_2}x_2-k_2) \end{array} \right.$$

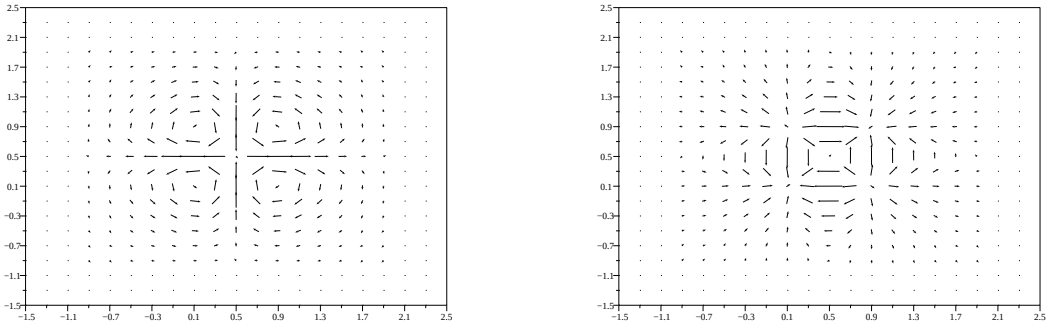


Figure 1: Examples of divergence-free (on the left) and curl-free (on the right) vector wavelets in dimension two.

This construction is extended to three dimensions in [9, 10].

We gave a similar construction for d-dimensional anisotropic divergence-free and gradient wavelets in [7].

2 Wavelet algorithms

2.1 Helmholtz decomposition

2.1.1 Principle of Helmholtz decomposition and the Leray projector

The Helmholtz decomposition [16, 3] consists in splitting a vector function $\mathbf{u} \in (L^2(\mathbb{R}^d))^d$ into its divergence-free component $\mathbf{u}_{\mathrm{div}}$ and a gradient vector. More precisely, there exist a potential-function p and a stream-function ψ such that:

$$\mathbf{u} = \mathbf{u}_{\text{div}} + \nabla p \quad \text{and} \quad \mathbf{u}_{\text{div}} = \mathbf{curl}\ \psi \tag{2.1}$$

Moreover, the functions $\mathbf{curl}\ \psi$ and ∇p are orthogonal in $(L^2(\mathbb{R}^d))^d$. The stream-function ψ and the potential-function p are unique, up to an additive constant.
In $\mathbb{R}^2$, the stream-function is a scalar valued function, whereas in $\mathbb{R}^3$ it is a 3D vector function. This decomposition may be viewed as the following orthogonal space splitting:

$$(L^2(\mathbb{R}^d))^d = \mathbf{H}_{\text{div},0}(\mathbb{R}^d) \oplus^{\perp} \mathbf{H}_{\text{curl},0}(\mathbb{R}^d) \tag{2.2}$$

where

$$\mathbf{H}_{\text{div},0}(\mathbb{R}^d) = \{\mathbf{v} \in (L^2(\mathbb{R}^d))^d / \text{div}\ \mathbf{v} \in L^2(\mathbb{R}^d), \quad \text{div}\ \mathbf{v} = 0\}$$

is the space of divergence-free vector functions, and

$$\mathbf{H}_{\text{curl},0}(\mathbb{R}^d) = \{\mathbf{v} \in (L^2(\mathbb{R}^d))^d / \mathbf{curl}\ \mathbf{v} \in (L^2(\mathbb{R}^d))^d, \quad \mathbf{curl}\ \mathbf{v} = 0\}$$

is the space of curl-free vector functions (if $d = 2$ we have to replace $\mathbf{curl}\ \mathbf{v} \in (L^2(\mathbb{R}^d))^d$ by $\text{curl}\ \mathbf{v} \in L^2(\mathbb{R}^2)$ in the definition). For the whole space $\mathbb{R}^d$, the proofs of the above decompositions can be derived easily by mean of the Fourier transform. In more general domains, we refer to [16, 3]. Notice that one can also prove that $\mathbf{H}_{\text{div},0}(\mathbb{R}^d)$ is the space of **curl** functions, whereas $\mathbf{H}_{\text{curl},0}(\mathbb{R}^d)$ is the space of gradient functions.
The objective now is to generate a wavelet-Helmholtz decomposition.

2.1.2 Iterative wavelet Helmhotz decomposition algorithm

Instead of the previous orthogonal direct sum (2.2), the divergence-free and gradient wavelet decompositions provide the following direct sums of vector spaces:

$$(L^2(\mathbb{R}^d))^d = \mathbf{H}_{\text{div},0} \oplus \mathbf{H}_{\text{n}}\ , \qquad (L^2(\mathbb{R}^d))^d = \mathbf{H}_{\text{N}} \oplus \mathbf{H}_{\text{curl},0}$$

We call these biorthogonal projectors P_{div} and Q_{curl}:

$$\mathbf{v} = \text{P}_{\text{div}}\, \mathbf{v} + \text{Q}_{\text{n}}\, \mathbf{v}\ , \qquad \mathbf{v} = \text{P}_{\text{N}}\, \mathbf{v} + \text{Q}_{\text{curl}}\, \mathbf{v}$$

Then, when we alternatively apply the divergence-free wavelet decomposition and then the curl-free wavelet decomposition, we define a sequence $(\mathbf{v}^p)_{p\in\mathbb{N}} \in \left(L^2(\mathbb{R}^d)\right)^d$:

$$\mathbf{v}^p = \underbrace{\text{P}_{\text{div}}\, \mathbf{v}^p}_{\mathbf{v}^p_{\text{div}}} + \underbrace{\text{Q}_{\text{curl}}\, \text{Q}_{\text{n}}\, \mathbf{v}^p}_{\mathbf{v}^p_{\text{curl}}} + \underbrace{\text{P}_{\text{N}}\, \text{Q}_{\text{n}}\, \mathbf{v}^p}_{\mathbf{v}^{p+1}}$$

Finally, as this sequence converges to 0 in L^2, we have:

$$\mathbf{v}_{\text{div}} = \sum_{p=0}^{+\infty} \mathbf{v}^p_{\text{div}} \qquad\qquad \mathbf{v}_{\text{curl}} = \sum_{p=0}^{+\infty} \mathbf{v}^p_{\text{curl}}$$

This algorithm is proved to converge in arbitrary dimension with Shannon wavelets, and we have the following convergence theorem proved in [10]:

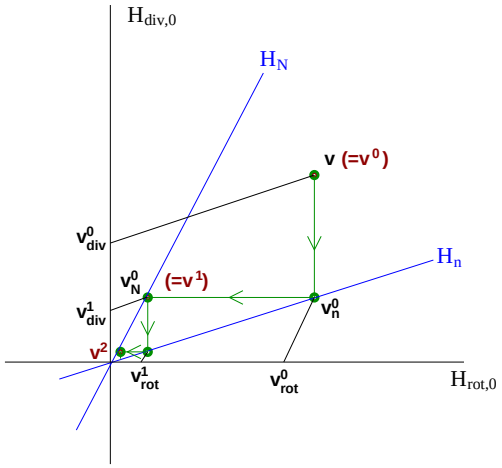


Figure 2: Construction of the sequences $\mathbf{v}^p_{\text{div}}$ and $\mathbf{v}^p_{\text{curl}}$, and idealistic schematization of the convergence process of the algorithm with $\mathbf{H}_{\text{n}} = \text{vect}\{\Psi^{\text{n}}_{\mathbf{j},\mathbf{k}}\}$ and $\mathbf{H}_{\text{N}} = \text{vect}\{\Psi^{\text{N}}_{\mathbf{j},\mathbf{k}}\}$.

Theorem 2.1 *Let* $\mathbf{v} \in \left(L^2(\mathbb{R}^d)\right)^d$*, and let the sequence* $(\mathbf{v}^p)_{p\geq 0}$ *be defined by:*

$$\mathbf{v}^0 = \mathbf{v} \qquad \text{and} \qquad \mathbf{v}^{p+1} = \mathrm{P}_N \mathrm{Q}_n \ \mathbf{v}^p, \quad p \geq 0 \tag{2.3}$$

where Q_n *and* P_N *are the complementary projectors associated respectively to divergence-free wavelets and curl-free wavelets. With Shannon wavelets, the sequence* $(\mathbf{v}^p)$ *satisfies, in* L^2 *norm:*

$$\|\mathbf{v}^p\| \leq \left(\frac{9}{16}\right)^p \|\mathbf{v}\|$$

Experimentally, we also observe the convergence for many kinds of 2D and 3D wavelets [9]. This is the algorithm we'll use in the numerical scheme for the solution of the incompressible Navier-Stokes equations.

2.2 Implicit Heat kernel

As we would like the numerical scheme to be sufficiently stable in time, we adopted an implicit resolution of the Heat kernel. This resolution is based on wavelet preconditionning of elliptic operators. It is related to the works of J.Liandrat and A.Cohen [26, 5]. For each component of the velocity $\mathbf{u}$, we apply this algorithm. Hence we consider only scalar functions.

We want to solve the equation :

$$(Id - \alpha\Delta)v = u, \qquad \alpha = \nu\delta t, \qquad u : \mathbb{R}^d \to \mathbb{R} \tag{2.4}$$

With a wavelet decomposition, u can be written in a basis $(\psi_{\mathbf{j},\mathbf{k}})$

$$u = \sum_{\mathbf{j},\mathbf{k}\in\mathbb{Z}^d} d_{\mathbf{j},\mathbf{k}}\psi_{\mathbf{j},\mathbf{k}}$$

we regroup the wavelets of a same level in $u_{\mathbf{j}} = \sum_{\mathbf{k}\in\mathbb{Z}^d} d_{\mathbf{j},\mathbf{k}}\psi_{\mathbf{j},\mathbf{k}}$, then

$$u = \sum_{\mathbf{j}\in\mathbb{Z}^d} u_{\mathbf{j}}$$

with $u_{\mathbf{j}}$ localized in frequency.

Let us say that $u_{\mathbf{j}}$ has frequency average $\omega_{\mathbf{j}}$ with $\omega_{\mathbf{j}}^2 \sim \sum_{i=1}^{d} 2^{2j_i}$.
We can solve $(Id - \alpha\Delta)v = u$, by taking:

$$v_0 = \sum_{\mathbf{j}\in\mathbb{Z}^d} v_{\mathbf{j},0}, \qquad \text{with} \qquad v_{\mathbf{j},0} = \frac{1}{1+\alpha\omega_{\mathbf{j}}^2} u_{\mathbf{j}}$$

which is a first approximation of the solution.
Then, let

$$v_{n+1} = v_n + \sum_{\mathbf{j}\in\mathbb{Z}^d} \frac{1}{1+\alpha\omega_{\mathbf{j}}^2}(u_{\mathbf{j}} - (Id - \alpha\Delta)v_{n\,\mathbf{j}})$$

The sequence (v_n) goes to v, the solution of equation (2.4). The smaller α is, the faster the algorithm converges. This algorithm is presented extensively in [8]. In that paper, we also prove the following convergence theorem:

Theorem 2.2 *Let u in $L^2(\mathbb{R}^d)$, and let the sequence $(v_n)_{n\geq 0}$ be defined by:*

$$v_0 = 0, \qquad v_{n+1} = v_n + \sum_{\mathbf{j}\in\mathbb{Z}^d} \frac{1}{1+\frac{5\pi}{2}\alpha(\sum_{i=1}^{d} 2^{2j_i})}(u_{\mathbf{j}} - (Id - \alpha\Delta)v_{n\,\mathbf{j}}) \tag{2.5}$$

With Shannon wavelets, the sequence (v_n) satisfies, in L^2 norm:

$$\|v_n - v\| \leq \left(\frac{3\alpha}{2\delta x^2 + 5\alpha}\right)^n \|v_0 - v\|$$

where v is the solution of equation (2.4) and δx the mesh size of the smallest computed scale.

As a result,

$$\|v_n - v\| \leq \left(\frac{3}{5}\right)^n \|v_0 - v\|$$

But if $\frac{\nu\delta t}{\delta x^2} \ll 1$ (remind that $\alpha = \nu\delta t$), then it is interesting to consider

$$\|v_n - v\| \leq \left(\frac{3\nu\delta t}{2\delta x^2}\right)^n \|v_0 - v\|$$

We apply this method with the divergence-free wavelet basis. However, with the linear operator $(Id - \alpha\Delta)^{-1}$, the divergence-free condition is preserved.

3 Numerical scheme for Navier-Stokes equations

In this Section, we present a divergence-free wavelet numerical method, for the solution of the incompressible Navier-Stokes equations. The Navier-Stokes equations, written in velocity-pressure formulation (without a forcing term) are given by:

$$\begin{cases} \frac{\partial \mathbf{u}}{\partial t} + (\mathbf{u}\cdot\nabla)\mathbf{u} + \nabla p - \nu\Delta\mathbf{u} = 0 \\ \nabla\cdot\mathbf{u} = 0 \end{cases} \tag{3.1}$$

The scheme for solving the Navier Stokes equations is defined by means of the wavelet algorithms previously described. We make a numerical analysis of this scheme and prove

its stability. This stability is unconditional for low Reynolds numbers and under a Courant-Friedrich-Lewy condition for high Reynolds numbers. The velocity $\mathbf{u}$ is expressed, at each step of the scheme in the divergence-free wavelet basis (anisotropic or generalized).

We denote by $\mathbb{P}$ the wavelet Leray projector. The time step is denoted by δt, and the smallest computed scale has mesh δx.

Remark 3.1 (Computation of the pressure) *The Helmholtz wavelet decomposition of the non-linear term* $(\mathbf{u}\cdot\nabla)\mathbf{u}$ *entails the pressure thanks the following relation:*

$$[(\mathbf{u}\cdot\nabla)\mathbf{u}]_{\rm curl} = (\mathbf{u}\cdot\nabla)\mathbf{u} - \mathbb{P}\left[(\mathbf{u}\cdot\nabla)\mathbf{u}\right] = -\nabla p = \sum_{\mathbf{j},\mathbf{k}} d_{{\rm curl}\,\mathbf{j},\mathbf{k}}\,\frac{1}{4}\,\nabla\left(\psi_1(2^{j_1}x_1-k_1)\psi_1(2^{j_2}x_2-k_2)\right)$$

Hence the pressure term is eliminated from the Navier-Stokes equations (3.1) after the Leray projection and will never appear in the numerical scheme or the computations. But it can be easily computed.

Written with the orthogonal projector $\mathbb{P}$, equations (3.1) become:

$$\left\{\begin{array}{l} \frac{\partial\mathbf{u}}{\partial t} + \mathbb{P}\left[(\mathbf{u}\cdot\nabla)\mathbf{u}\right] - \nu\Delta\mathbf{u} = 0 \\ \nabla\cdot\mathbf{u} = 0 \end{array}\right. \tag{3.2}$$

3.1 Semi-implicit Euler scheme

By "semi-implicit" we mean that we integrate the Heat kernel Δ implicitly in time, thanks to the wavelet algorithm defined in Section 2.2, and we treat the non-linear term $(\mathbf{u}\cdot\nabla)\mathbf{u}$ explicitely. Additionally, we compute $\mathbb{P}\left[(\mathbf{u}\cdot\nabla)\mathbf{u}\right]$ using the algorithm of Section 2.1.2. Clearly, all the results given by these algorithms are expressed in terms of divergence-free wavelet coefficients.

The semi-implicit Euler scheme for the Navier-Stokes equations is given by:

$$(Id - \nu\delta t\Delta)\,\mathbf{u}_{n+1} = \mathbf{u}_n - \delta t\mathbb{P}\left[(\mathbf{u}_n\cdot\nabla)\mathbf{u}_n\right] \tag{3.3}$$

3.2 The L^2-stability

We assume that the solution $\mathbf{u}(t,\mathbf{x})$ of the Navier-Stokes equations (3.2) is in $\mathbb{R}^d$ for $d=2$ or 3, with initial data $\mathbf{u}_0$, and for $t\in[0,T]$. We also assume that $\mathbf{u}$ is continuously differentiable in space and that $\mathbf{u}$ and its derivatives go to zero at infinity ($\|\mathbf{x}\|\to+\infty$). Hence:

$$A_0 = \sup_{\mathbf{x}\in\mathbb{R}^d,\,t\in[0,T]}\|\mathbf{u}(t,\mathbf{x})\| < +\infty$$

$$A_1 = \sup_{\mathbf{x}\in\mathbb{R}^d,\,t\in[0,T]}\|\nabla\mathbf{u}(t,\mathbf{x})\| < +\infty$$

For studying stability, we assume that at step n, we have a small error ε_n. Hence, instead of having exactly $\mathbf{u}_n$, we have $\mathbf{u}_n+\varepsilon_n$. Then, at step $n+1$, the error with respect to $\mathbf{u}((n+1)\delta t,\cdot)$, is due, on one hand to the newly introduced error in the algorithm (which is called the consistency error), and on the other hand, to the increase of the error coming from the previous steps (and which is concerned with stability).

If we prove that there exists a constant C (which will depend on A_0 and A_1) such that $\|\varepsilon_{n+1}\|\le(1+C\delta t)\|\varepsilon_n\|$, then the scheme is stable. In our case, we recover the L^2-stability thanks to the following well-known result:

Lemma 3.1 *Let* $\mathbf{u},\mathbf{v}\in H^1(\mathbb{R}^d)^d$, H^1 *denoting the Sobolev space, be such that* $(\mathbf{u}\cdot\nabla)\mathbf{v}\in L^2$. *If* $\mathbf{u}\in\boldsymbol{H}_{{\rm div},0}(\mathbb{R}^d)$, *then* $\mathbf{v}\perp(\mathbf{u}\cdot\nabla)\mathbf{v}$ *for the* L^2 *scalar product, i.e.*

$$<\mathbf{v},(\mathbf{u}\cdot\nabla)\mathbf{v}>_{L^2} = \int_{\mathbf{x}\in\mathbb{R}^d}\mathbf{v}\cdot(\mathbf{u}\cdot\nabla)\mathbf{v}\,d\mathbf{x} = 0$$

Proof:

$$\begin{aligned}
< \mathbf{v}, (\mathbf{u}\cdot\nabla)\mathbf{v} > &= \int_{\mathbf{x}\in\mathbb{R}^d} \mathbf{v}\cdot(\mathbf{u}\cdot\nabla)\mathbf{v}\, d\mathbf{x} \\
&= \int_{\mathbf{x}\in\mathbb{R}^d} \sum_{i=1}^d v_i(\mathbf{x}) \sum_{k=1}^d u_k(\mathbf{x})\partial_k v_i(\mathbf{x})\, d\mathbf{x} \\
&= \int_{\mathbf{x}\in\mathbb{R}^d} \sum_{k=1}^d u_k(\mathbf{x}) \left(\sum_{i=1}^d v_i(\mathbf{x})\partial_k v_i(\mathbf{x})\right) d\mathbf{x} \\
&= \int_{\mathbf{x}\in\mathbb{R}^d} \sum_{k=1}^d u_k(\mathbf{x})\partial_k \left(\tfrac{1}{2}\sum_{i=1}^d v_i^2(\mathbf{x})\right) d\mathbf{x} \\
&= -\int_{\mathbf{x}\in\mathbb{R}^d} \left(\sum_{k=1}^d \partial_k u_k(\mathbf{x})\right) \left(\tfrac{1}{2}\sum_{i=1}^d v_i^2(\mathbf{x})\right) d\mathbf{x} \\
&= 0
\end{aligned}$$

Here, we used integration by parts and the fact that $\operatorname{div}\mathbf{u} = \sum_{k=1}^d \partial_k u_k = 0$.

Remark 3.2 *This result is still valid on an open set Ω with $\mathbf{u} = 0$ on the boundaries, or even just slipping conditions ($\mathbf{u}\cdot\mathbf{n} = 0$).*

Remark 3.3 *This result also yields orthogonality of the vectors $\mathbf{v}$ and $(\mathbf{v}\cdot\nabla)\mathbf{v}$ for $\mathbf{v} \in \boldsymbol{H}_{\mathrm{div},0}(\mathbb{R}^d)$.*

Theorem 3.1 *An error ε_t equal to ε at time $t = 0$ propagates in the wavelet numerical scheme (3.3) bounded by:*

$$\|\varepsilon_t\|_{L^2} \leq e^{(\frac{A_0^2}{2}\frac{\delta t}{\delta x^2} + A_1)t} \|\varepsilon\|_{L^2}$$

where A_0 and A_1 are constants depending on $\mathbf{u}$ introduced at the beginning of this section, δt the time step and δx the mesh of the smallest computed scale.

Then the scheme (3.3) is stable under the CFL condition $\frac{\delta t}{\delta x^2} \leq C$ for a fixed constant $C > 0$ which can be chosen equal to $\frac{2A_1}{A_0^2}$ for instance.

Proof: If we take into account the propagation of the error ε_n along with $\mathbf{u}_n$, then the scheme (3.3) becomes:

$$(Id - \nu\delta t\Delta)(\mathbf{u}_{n+1} + \varepsilon_{n+1}) = \mathbf{u}_n + \varepsilon_n - \delta t\mathbb{P}\left[((\mathbf{u}_n + \varepsilon_n)\cdot\nabla)(\mathbf{u}_n + \varepsilon_n)\right] \tag{3.4}$$

As a result ε_n satisfies:

$$(Id - \nu\delta t\Delta)\varepsilon_{n+1} = \varepsilon_n - \delta t\mathbb{P}\left[(\varepsilon_n\cdot\nabla)\mathbf{u}_n + (\mathbf{u}_n\cdot\nabla)\varepsilon_n + (\varepsilon_n\cdot\nabla)\varepsilon_n\right] \tag{3.5}$$

As $\varepsilon_n \in \mathbf{H}_{\mathrm{div},0}(\mathbb{R}^d)$, it is orthogonal to all gradient functions, and so

$$\varepsilon_n \perp (\mathbf{u}_n\cdot\nabla)\varepsilon_n \quad \Longrightarrow \quad \varepsilon_n \perp \mathbb{P}\left[(\mathbf{u}_n\cdot\nabla)\varepsilon_n\right]$$

and

$$\varepsilon_n \perp (\varepsilon_n\cdot\nabla)\varepsilon_n \quad \Longrightarrow \quad \varepsilon_n \perp \mathbb{P}\left[(\varepsilon_n\cdot\nabla)\varepsilon_n\right]$$

Then, with $\eta_n = \mathbb{P}\left[(\mathbf{u}_n\cdot\nabla)\varepsilon_n + (\varepsilon_n\cdot\nabla)\varepsilon_n\right]$,

$$\|\varepsilon_n + \eta_n\delta t\|_{L^2}^2 = \|\varepsilon_n\|_{L^2}^2 + \|\eta_n\|_{L^2}^2\delta t^2$$

and

$$\|\varepsilon_n + \eta_n \delta t\|_{L^2} = \|\varepsilon_n\|_{L^2} + \frac{\|\eta_n\|_{L^2}^2}{2\|\varepsilon_n\|_{L^2}} \delta t^2 + o(\delta t^2)$$

where we noted $o(h)$, a quantity going to 0 faster than h for $\delta t \to 0$ and $\delta x \to 0$.
For the stability, we can assume that ε_n remains in our discretization space. Indeed, the error introduced by the truncation of $\mathbf{u}_n$ is concerned with the consistency error. It accumulates but it is not amplified by the numerical scheme. Hence, we have the following bounds:

$$\|\eta_n\|_{L^2} \leq \|\mathbf{u}_n\|_{L^\infty} \|\varepsilon_n\|_{H^1} + \|\varepsilon_n\|_{L^\infty} \|\varepsilon_n\|_{H^1}$$

$$\|\varepsilon_n\|_{H^1} \leq \frac{\|\varepsilon_n\|_{L^2}}{\delta x}$$

$$\|\varepsilon_n\|_{L^\infty} \leq \frac{\|\varepsilon_n\|_{L^2}}{\delta x^{d/2}}$$

$$\|\mathbf{u}_n\|_{L^\infty} \leq A_0$$

we obtain:

$$\frac{\|\eta_n\|_{L^2}^2}{2\|\varepsilon_n\|_{L^2}} \leq \frac{A_0^2 \|\varepsilon_n\|_{L^2}}{2\delta x^2} + A_0 \frac{\|\varepsilon_n\|_{L^2}^2}{\delta x^2} + \frac{\|\varepsilon_n\|_{L^2}^3}{\delta x^{2+d/2}}$$

as well as:

$$\|\delta t \mathbb{P}\left[(\varepsilon_n \cdot \nabla)\mathbf{u}_n\right]\| \leq \|\varepsilon_n\|_{L^2} \|\nabla \mathbf{u}_n\|_{L^\infty} \delta t \leq A_1 \|\varepsilon_n\|_{L^2} \delta t$$

And finally:

$$\|\left(Id - \nu \delta t \Delta\right) \varepsilon_{n+1}\|_{L^2} \leq \left(1 + \frac{A_0^2}{2} \frac{\delta t^2}{\delta x^2} + A_0 \frac{\|\varepsilon_n\|_{L^2} \delta t^2}{\delta x^2} + \frac{\|\varepsilon_n\|_{L^2}^2}{2\delta x^{2+d}} \delta t^2 + A_1 \delta t\right) \|\varepsilon_n\|_{L^2}$$

As we can assume that $\|\varepsilon_n\|_{L^2} = o(1)$ and even $\|\varepsilon_n\|_{L^2} = o(\delta x^{d/2})$, for $d = 2, 3$ (these apriori estimates will be verified latter), this gives

$$\|\left(Id - \nu \delta t \Delta\right) \varepsilon_{n+1}\|_{L^2} \leq \left(1 + (\frac{A_0^2}{2} \frac{\delta t}{\delta x^2} + A_1) \delta t\right) \|\varepsilon_n\|_{L^2}$$

Any role played by the implicit Laplacian would be favorable for the L^2-stability since $\|\varepsilon_{n+1}\|_{L^2} \leq \|\left(Id - \nu \delta t \Delta\right) \varepsilon_{n+1}\|_{L^2}$. But, assume that the term $(Id - \nu \delta t \Delta)$ doesn't play any role because $\nu \frac{\delta t}{\delta x^2} \ll 1$. Then we have the following **CFL condition**:

$$\frac{\delta t}{\delta x^2} \leq C \tag{3.6}$$

for some $C > 0$ constant. Or, if we want something more *precise*:

$$\frac{A_0^2}{2} \frac{\delta t}{\delta x^2} = A_1 \tag{3.7}$$

Remark 3.4 *This stability condition can be compared with the usual CFL condition $\delta t \leq C \delta x$ which is much weaker and more interesting. In practice, we observe a stability condition $\delta t \leq C \delta x^2$ for the Euler scheme (3.3) which confirm the theoretical CFL condition (3.6) of order 2. For the order two Adam-Bashford divergence-free wavelet scheme (3.10) we observe an empirical stability condition $\delta t \leq C \delta x^{1.3}$.*

The stability is then guaranteed since, for an initial error $\varepsilon \in L^2$ at time 0,

$$\|\varepsilon_{T/\delta t}\|_{L^2} \leq \left(1 + (\frac{A_0^2}{2}\frac{\delta t}{\delta x^2} + A_1)\delta t\right)^{T/\delta t} \|\varepsilon\|_{L^2} \leq e^{(\frac{A_0^2}{2}\frac{\delta t}{\delta x^2} + A_1)T}\|\varepsilon\|_{L^2}$$

Remark 3.5 *If we had just*

$$\varepsilon_{n+1} = \varepsilon_n + \delta t \eta_n$$

with no orthogonality between ε_n and η_n, then the stability would not hold since

$$\|\varepsilon_{n+1}\|_{L^2} \sim \|\varepsilon_n\|_{L^2} + \|\eta_n\|_{L^2}\delta t \sim (1 + \frac{A_0}{\delta x}\delta t)\|\varepsilon_n\|_{L^2}$$

and then

$$\|\varepsilon_{T/\delta t}\|_{L^2} \sim e^{(\frac{A_0}{\delta x})T}\varepsilon_0$$

which goes exponentially to infinity for $\delta x \to 0$.

Now, we take the implicit Laplacian into account. Assume we applied first the implicit Laplacian and then the convection term, then

$$\mathbf{u}_{n+1} = (Id - \nu\delta t\Delta)^{-1}\mathbf{u}_n - \delta t\mathbb{P}\left[\left((Id - \nu\delta t\Delta)^{-1}\mathbf{u}_n \cdot \nabla\right)(Id - \nu\delta t\Delta)^{-1}\mathbf{u}_n\right]$$

and so

$$\begin{aligned}\varepsilon_{n+1} = (Id - \nu\delta t\Delta)^{-1}\varepsilon_n - \delta t\mathbb{P}\Big[&\left((Id - \nu\delta t\Delta)^{-1}\varepsilon_n \cdot \nabla\right)(Id - \nu\delta t\Delta)^{-1}\mathbf{u}_n \\ &+\left((Id - \nu\delta t\Delta)^{-1}\mathbf{u}_n \cdot \nabla\right)(Id - \nu\delta t\Delta)^{-1}\varepsilon_n\Big]\end{aligned}$$

We neglected the last term $(\varepsilon_n \cdot \nabla)\varepsilon_n$ in comparison with equation (3.5) since it doesn't play any rôle. As $(Id - \nu\delta t\Delta)^{-1}\varepsilon_n$ is divergence-free, we have the same orthogonalities as in (3.5). We replace the bound for $\|\varepsilon_n\|_{H^1}$ by:

$$\|\nabla(Id - \nu\delta t\Delta)^{-1}\varepsilon_n\|_{L^2} \leq \sup_{\delta x \leq \alpha < +\infty}\left(1 + \frac{\nu\delta t}{\alpha^2}\right)\frac{\|\varepsilon_n\|_{L^2}}{\alpha} \leq \frac{\|\varepsilon_n\|_{L^2}}{2\sqrt{\nu\delta t}}$$

and we have

$$\|\varepsilon_{n+1}\|_{L^2} \leq (1 + (\frac{A_0^2}{8\nu} + A_1)\delta t)\|\varepsilon_n\|_{L^2} \tag{3.8}$$

Hence we have unconditional stability. But one has to remark that this is interesting only if $\frac{A_0^2}{8\nu} \leq A_1$, that is for low Reynolds numbers.

To summarize, the stability conditions are:

$$A_0^2 \leq 8\nu A_1 \qquad \text{or} \qquad \delta t \leq C\delta x^2$$

with $C > 0$ a constant. And with condition (3.7) $A = A_1 = \frac{A_0^2}{2}\frac{\delta t}{\delta x^2}$, the L^2-stability of the implicit Euler scheme (3.3) is given by:

$$\|\varepsilon_{n+1}\|_{L^2} \leq (1 + 2A\delta t)\|\varepsilon_n\|_{L^2} \tag{3.9}$$

3.3 An order two scheme

In the numerical experiments of Section 5, we tested a scheme of order two in time. It's derived from an Adams-Bashford sheme of order 2 for the non-linear term. This scheme proceeds with an intermediate step $\mathbf{u}_{n+1/2}$ as follows:

$$\left(Id - \nu\frac{\delta t}{2}\Delta\right)\mathbf{u}_{n+1/2} = \mathbf{u}_n - \frac{\delta t}{2}\mathbb{P}\left[(\mathbf{u}_n\cdot\nabla)\mathbf{u}_n\right]$$

and then

$$\left(Id - \nu\frac{\delta t}{2}\Delta\right)\mathbf{u}_{n+1} = \mathbf{u}_n + \delta t\left(\frac{\nu}{2}\Delta\mathbf{u}_n - \mathbb{P}\left[(\mathbf{u}_{n+1/2}\cdot\nabla)\mathbf{u}_{n+1/2}\right]\right) \tag{3.10}$$

As $\mathbf{u}_{n+1/2}$ is assymptotically close to $\mathbf{u}_n$, the same arguments as in Section 3.1 can be used for proving stability.

4 The adaptive scheme

At each step of the semi-implicit Euler scheme (3.3) and the order two scheme, the solution $\mathbf{u}_n$ and the non-linear term $\mathbb{P}\left[(\mathbf{u}_n\cdot\nabla)\mathbf{u}_n\right]$ are expressed in terms of divergence-free wavelets. The only basic algorithm we use is the Fast Wavelet Transform.

We'll sketch an adaptive method for the *numerical resolution* of incompressible Navier-Stokes equations in dimensions 2 and 3:

$$\begin{cases} \frac{\partial\mathbf{u}}{\partial t} + \mathbf{u}.\nabla\mathbf{u} + \nabla p = \nu\Delta\mathbf{u} + \mathbf{f} & t\in[0,T],\ x\in\mathbb{R}^d,\ d=2 \text{ or } 3 \\ \text{div } \mathbf{u} = \nabla\cdot\mathbf{u} = \sum_{i=1}^n \frac{\partial u_i}{\partial x_i} = 0 & \\ \mathbf{u}(x,0) = \mathbf{u}_0(x) & \end{cases}$$

The adaptivity is based on the wavelet discretization. At each time step n, we first find a set Λ_n of active wavelet coefficients. The other coefficients are ignored and set to zero, or in a really adaptive scheme, not represented. Then the computed solution $\mathbf{u}_N$ is given by:

$$\mathbf{u}_N(n\delta t, x) = \sum_{\lambda\in\Lambda_n} c_{n,\lambda}\ \Psi^{\text{div}}_\lambda(x)$$

with $\#(\Lambda_n) = N$ (the set Λ_n has N elements) and $\Psi^{\text{div}}_\lambda \in \mathrm{H}_{\text{div},0} = \{\mathbf{u}\in L^2,\ \text{div}(\mathbf{u}) = 0\}$.

Further, we used two criteria for an element λ to be in Λ_n. Let

$$\mathbf{u}_n = \sum_\lambda c_{n,\lambda}\ \Psi^{\text{div}}_\lambda$$

and

$$\mathbb{P}\left[(\mathbf{u}_n\cdot\nabla)\mathbf{u}_n - \mathbf{f}\right] = \sum_\lambda d_{n,\lambda}\ \Psi^{\text{div}}_\lambda$$

with an L^2-normalization for the wavelets $\Psi^{\text{div}}_\lambda$. And let $\sigma_{0\,n} > 0$ and $\sigma_{1\,n} > 0$ be two thresholds.

The index λ is activated if $|c_{n,\lambda}| \geq \sigma_{0\,n}$ or $|d_{n,\lambda}| \geq \sigma_{1\,n}$. The first threshold $\sigma_{0\,n}$ allows to stock valuable information on $\mathbf{u}_N$, the second one $\sigma_{1\,n}$ takes into account the changes in the coefficients.

5 Numerical experiments

The experiment on which we test our algorithm, is called the "merging of 3 vortices". This experiment, without any forcing term, was originally designed by M. Farge and N. Kevlahan [30], and is often used to test new numerical methods [2, 17]. In order to provide a reference solution, the experiment of [2] was reproduced by using a pseudo-spectral method, solving the Navier-Stokes equations in velocity-pressure formulation.

The initial state is displayed in Figure 3 left. In the periodic box $[0,1]^2$, three vortices with a gaussian vorticity profile $\omega(\mathbf{x}) = \pi e^{-4\pi^4(x_1^2+x_2^2)}$ are present:

- one centered at $(3/8, 1/2)$ with amplitude 1,
- one centered at $(5/8, 1/2)$ also with amplitude 1,
- and one centered at $(5/8, 1/2 + \sqrt{2}/8)$ with amplitude $-1/2$.

The negative vortex is here to force the merging of the two positive ones. The time step was $\delta t = 10^{-2}$ and the viscosity $\nu = 5.10^{-5}$. The solution is computed on a 512×512 grid.

The vorticity fields at times $t = 0, 10, 20$ and 40 are displayed on Figure 3. The second row of Figure 3 displays the absolute values of the isotropic divergence-free wavelet coefficients of the *velocity field* at corresponding times, with an L^∞-normalization.

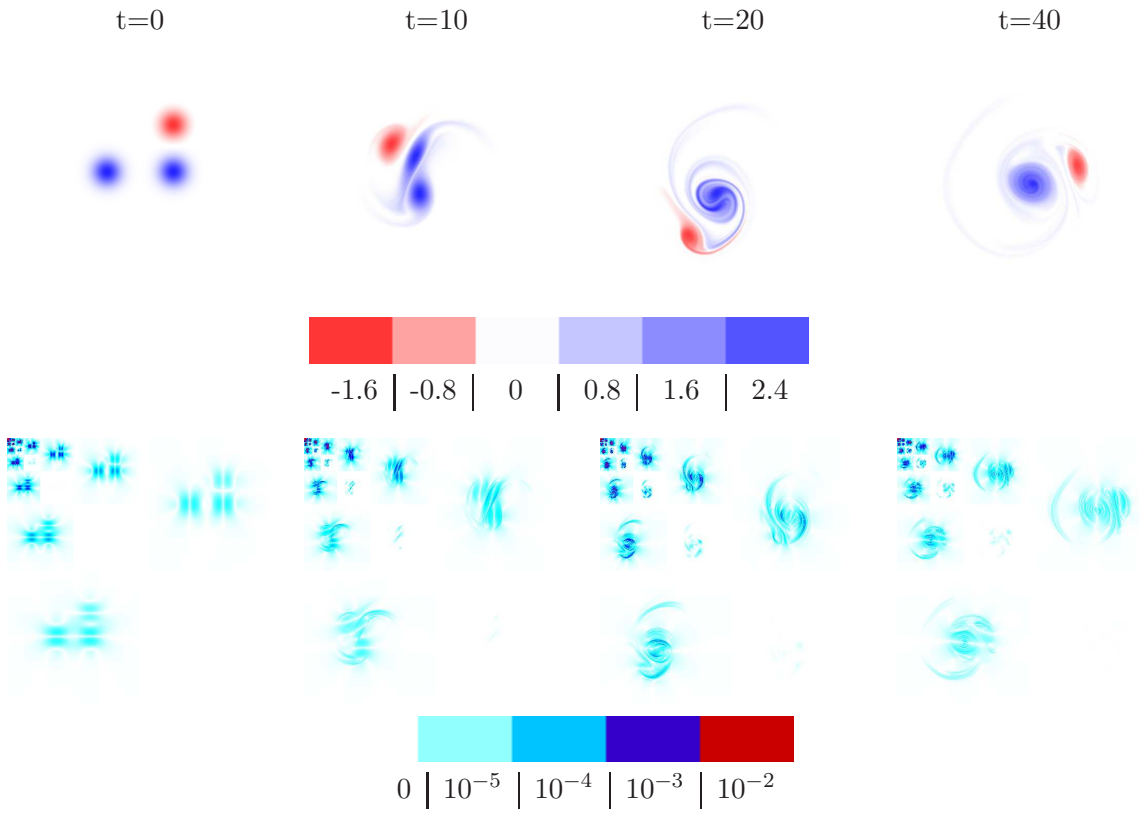
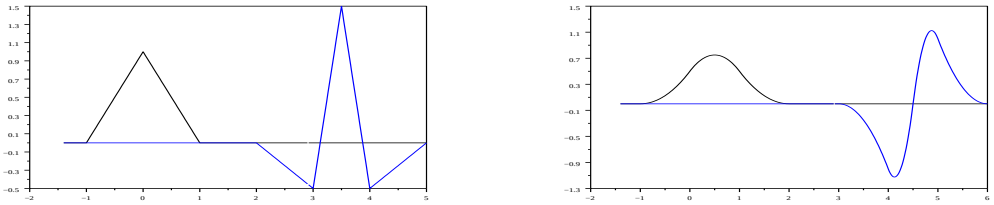


Figure 3: Vorticity fields at times $t = 0, 10, 20$ and 40, and corresponding isotropic divergence-free wavelet coefficients of the velocity, for the reference solution in pseudo-spectral code, on a 512^2 grid.

5.1 Full code

Here, we show the results of simulation of the "merging of three vortices" with the anisotropic divergence-free wavelet scheme of Section 3.3.

We used the spline wavelets of order 2 and 3 represented in Figure 4.



ϕ_0 ψ_0 ϕ_1 ψ_1

Figure 4: Scaling spline functions and associated spline wavelets of order 2 and 3 related by differentiation: $\psi_1' = 4\psi_0$.

The solution was computed on a 256^2 grid with $\delta t = 0.02$ and $\nu = 5.10^{-5}$. There were no thresholding. In order to make the algorithms of Sections 2.1.2 and 2.2 give a sufficient accuracy, we needed 7 iterations for the wavelet Helmholtz decomposition of $(\mathbf{u}_n \cdot \nabla)\mathbf{u}_n$, using the result of the previous step as an initial guess, and 3 iterations for the implicit Heat kernel wavelet solver.

t=0 t=10 t=20 t=40

Figure 5: Simulation of the "merging of three vortices" with the full wavelet code, using anisotropic wavelets in 256^2.

The results are represented in Figure 5 and are close to the reference solution in Figure 3. This code uses **uniquely** wavelet transforms.

We also represented the evolution of the set of wavelet coefficients, observing the ratio of coefficients (in L^2-normalization) which are above some fixed thresholds on Figure 6, for the "merging of three vortices". We took $\varepsilon = \sup(|c_{0,\lambda}|)$, the maximal value of the wavelet coefficients at time 0. And, we fixed the thresholds at $\varepsilon/4^j$ for $1 \leq j \leq 7$. Then the lowest threshold represented is $\varepsilon/16,384$.

One can observe that for this experiment, the complexity of the flow structure increases until reaching a maximum at time $t = 22$, and then decreases slowly.

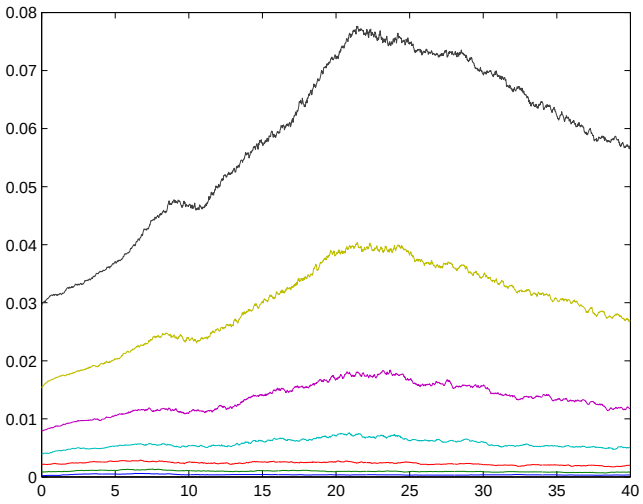


Figure 6: Evolution of the ratio of anisotropic divergence-free wavelet coefficients of the solution $\mathbf{u}_n$ of the "merging of three vortices" above thresholds equal to $\varepsilon/4^j$ for $1 \leq j \leq 7$.

5.2 Anisotropic pseudo-adaptive code

Here we investigate how the numerical wavelet scheme behaves with thresholding. As the implementation of a fully adaptive scheme is heavy, we just put to zero the wavelet coefficients for which we had neither $|c_{n,\lambda}| \geq \sigma_0$ nor $|d_{n,\lambda}| \geq \sigma_1$ as explained in Section 4, in the full anisotropic divergence-free wavelet code.

We represented the result of the thresholding in Figure 7. We can infer the number of coefficients verifying both thresholding conditions by substracting the area between the upper and middle curve from the lower curve.

The solution obtained in Figure 8 may be compared with Figures 3 and 5.

We also represented the computed L^2 relative error between this numerical solution and the reference solution of Figure 3. It is also compared with the error obtained for a pseudo-spectral code with the same number of grid points (256^2). One can notice that neither the use of wavelets, neither the thresholding destroyed the accuracy of the solution.

The *effects of the anisotropy* begin to be visible for higher thresholdings.

In Figure 11, they are already clearly present without destroying the solution. And the evolution of the thresholding represented on Figure 10, gives satisfactory results. But in Figure 12 with a higher threshold, when keeping a maximum of 4.5% of the coefficients, the final result is not admissible.

5.3 Generalized pseudo-adaptive code

We remedied the problem of anisotropy induced by thresholding, by using generalized divergence-free wavelets which are a mix between isotropic divergence-free wavelets and anisotropic divergence-free wavelets. These wavelets were first introduced in [7] and extensively treated in [11].

Here, we display their construction in dimension two.

Let $m \in \mathbb{N}$. The generalized divergence-free wavelets are composed of the following vector

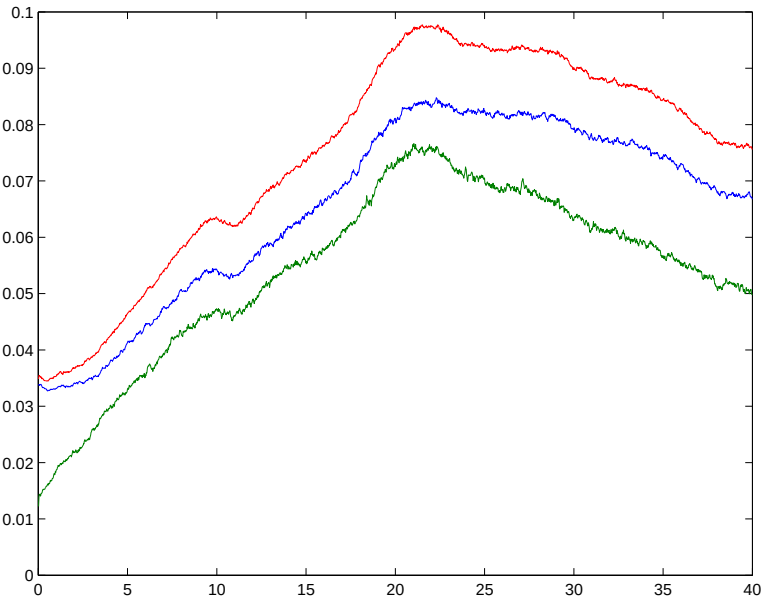


Figure 7: Ratio of coefficients above the threshold: the lowest curve is for the coefficients from the non-linear term, the middle curve from the solution $\mathbf{u}_n$, and the upper curve is the union of these two.

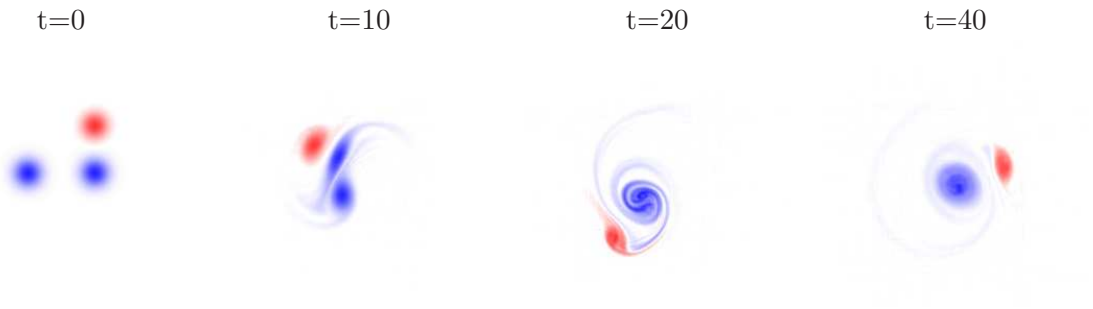


Figure 8: Simulation of the "merging of three vortices" by a wavelet code with thresholding corresponding to Figure 7, and using anisotropic wavelets in 256^2.

wavelets:

- $\Psi^{\mathrm{div}\,(1,1)}_{\mathbf{j},\mathbf{k}} = \left| \begin{array}{l} 2^{j_2}\psi_1(2^{j_1}x_1-k_1)\psi_0(2^{j_2}x_2-k_2) \\ -2^{j_1}\psi_0(2^{j_1}x_1-k_1)\psi_1(2^{j_2}x_2-k_2) \end{array} \right.$ with $|j_1-j_2|\le m$,
- $\Psi^{\mathrm{div}\,(1,0)}_{\mathbf{j},\mathbf{k}} = \left| \begin{array}{l} 2^{j_2-2}\psi_1(2^{j_1}x_1-k_1)(\varphi_0(2^{j_2}x_2-k_2)-\varphi_0(2^{j_2}x_2-k_2-1)) \\ -2^{j_1}\psi_0(2^{j_1}x_1-k_1)\varphi_1(2^{j_2}x_2-k_2) \end{array} \right.$ with $j_2=j_1-m$,
- $\Psi^{\mathrm{div}\,(0,1)}_{\mathbf{j},\mathbf{k}} = \left| \begin{array}{l} 2^{j_2}\varphi_1(2^{j_1}x_1-k_1)\psi_0(2^{j_2}x_2-k_2) \\ -2^{j_1-2}(\varphi_0(2^{j_1}x_1-k_1)-\varphi_0(2^{j_1}x_1-k_1-1))\psi_1(2^{j_2}x_2-k_2) \end{array} \right.$ with $j_1=j_2-m$.

where $\psi_1' = 4\psi_0$ and $\varphi_1' = \varphi_0(\,.\,) - \varphi_0(\,.\,-1)$.

The corresponding complement functions used for the divergence-free wavelet transform are:

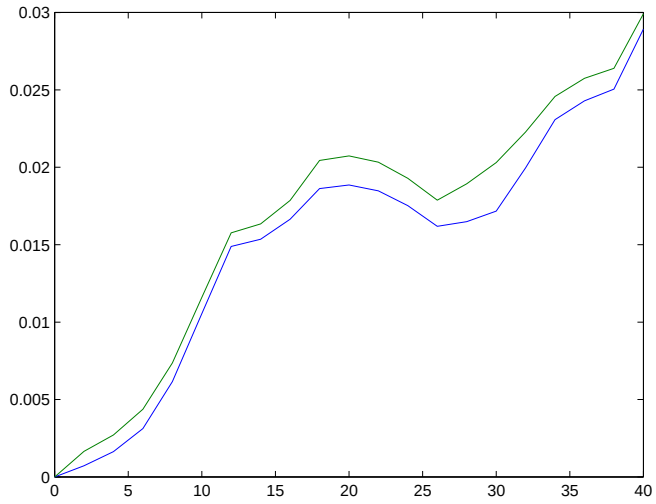


Figure 9: L^2 relative error for pseudo-spectral code (lower curve) and wavelet thresholded code (upper curve), corresponding to Figure 7, on a 256^2 grid, compared with the reference solution.

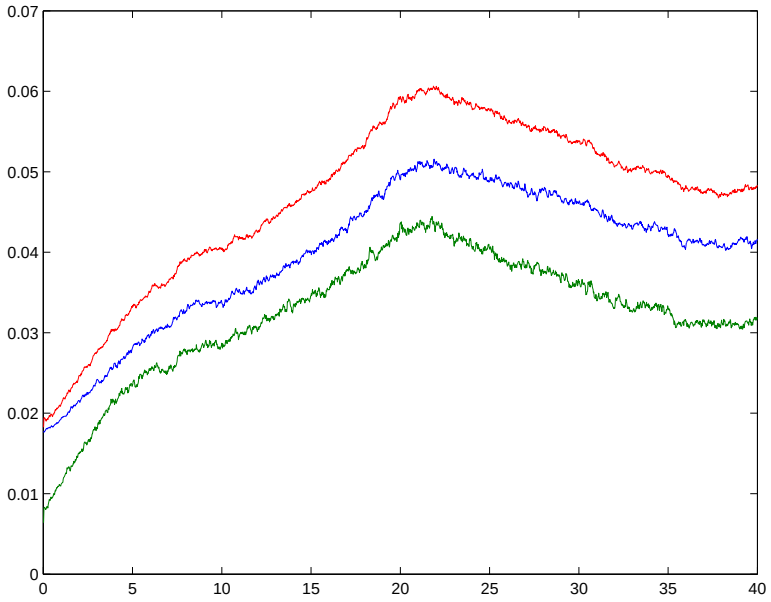


Figure 10: Ratio of coefficients kept for the computation of the solution $\mathbf{u}_n$ with an average thresholding.

- $\Psi^{\mathrm{n}\,(1,1)}_{\mathbf{j},\mathbf{k}} = \left| \begin{array}{l} 2^{j_1}\psi_1(2^{j_1}x_1 - k_1)\psi_0(2^{j_2}x_2 - k_2) \\ 2^{j_2}\psi_0(2^{j_1}x_1 - k_1)\psi_1(2^{j_2}x_2 - k_2) \end{array} \right.$ with $|j_1 - j_2| \leq m$,
- $\Psi^{\mathrm{div}\,(1,0)}_{\mathbf{j},\mathbf{k}} = \left| \begin{array}{l} \psi_1(2^{j_1}x_1 - k_1)\varphi_0(2^{j_2}x_2 - k_2) \\ 0 \end{array} \right.$ with $j_2 = j_1 - m$,

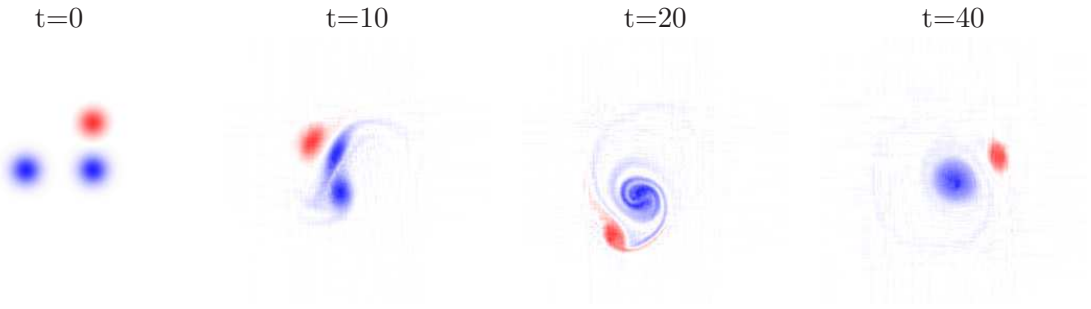


Figure 11: Simulation of the "merging of three vortices" by a wavelet code with an average thresholding corresponding to Figure 10, on a 256^2 grid, using anisotropic divergence-free wavelets.

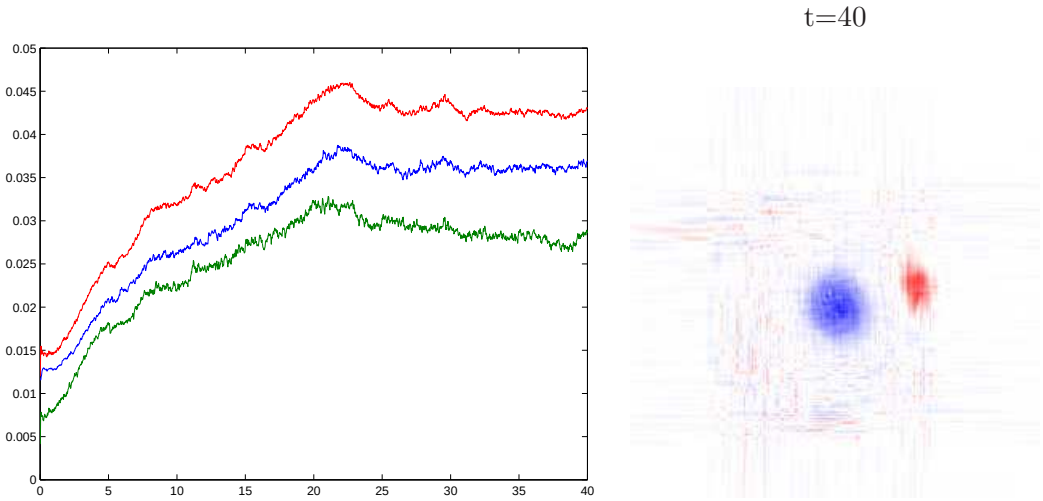


Figure 12: Ratio of active coefficients for a high thresholding (on the left), and final result at $t = 40$ (on the right), on a 256^2 grid, using anisotropic divergence-free wavelets.

- $\Psi^{\mathrm{div}\,(0,1)}_{\mathbf{j},\mathbf{k}} = \left| \begin{array}{l} 0 \\ \varphi_0(2^{j_1}x_1 - k_1)\psi_1(2^{j_2}x_2 - k_2) \end{array} \right.$ with $j_1 = j_2 - m$.

For $m = 0$, these wavelets are the usual isotropic divergence-free wavelets.

Contrarily to anisotropic wavelets, these wavelets do not lengthen and permit to refine the grid locally. With the generalized divergence-free wavelets, the algorithms of Sections 2.1.2 and 2.2 give still good results.

While the Helmholz wavelet decomposition of Section 2.1.2 doesn't converge for $m = 0$, it does converge for $m = 1$. In fact, for $m = 1$, wavelets $\Psi^{\mathrm{div}\,(1,0)}_{\mathbf{j},\mathbf{k}}$ and $\Psi^{\mathrm{div}\,(0,1)}_{\mathbf{j},\mathbf{k}}$ are better localized in frequency.

Hence, we sacrify a little part of space localization to get a sufficiently good localization in frequency, and assure the convergence of the Helmholtz algorithm of Section 2.1.2.

Even with a rather high threshold as in Figure 13, the quality of the solution represented in Figure 14 is quite good. And the thresholding evolves normally.

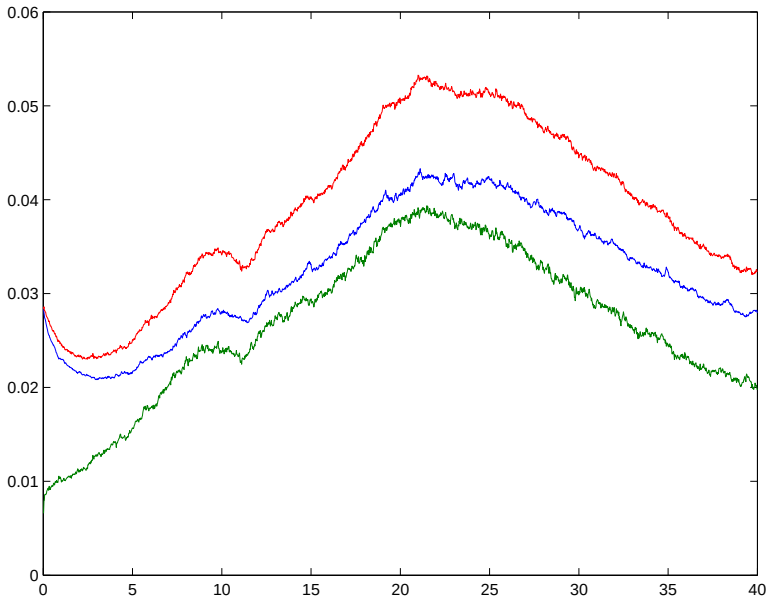


Figure 13: Ratio of activated wavelet coefficients in the case of generalized divergence-free wavelets on a 256^2 grid.

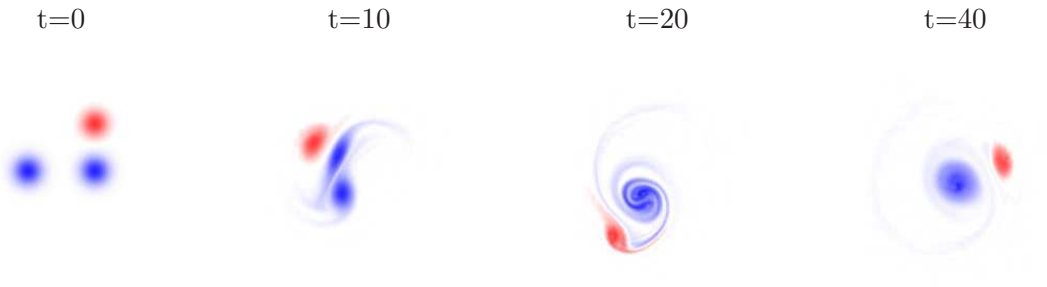


Figure 14: Simulation of the "merging of three vortices" by a generalized divergence-free wavelet code with thresholding corresponding to Figure 13, on a 256^2 grid.

Conclusion

To our knowledge, a numerical divergence-free wavelet method for solving Navier-Stokes equations was presented for the first time in this paper. The proposed method only relies on the Fast Wavelet Transform algorithm and is perfectly fitted for adaptivity.

Untill now, the use of divergence-free wavelets was limited to Galerkin methods applied to the Driven-Cavity-Stokes problem [32] or the equations of electromagnetics [35]. This limitation was due to the inexistence of divergence-free wavelet algorithms dealing with the non-linear term $(\mathbf{u} \cdot \nabla)\mathbf{u}$. The invention of the anisotropic divergence-free wavelets and more specifically of the generalized divergence-free wavelets [7] enables such algorithm (see Section 2.1.2 or [11]).

The numerical stability of this numerical scheme is shown under a CFL condition, and is mainly due to the use of the divergence-free wavelets that permit to verify exactly the

divergence-free condition ($\operatorname{div} \mathbf{u} = 0$).

An extensive numerical test with the experiment of the "merging of three vortices" is displayed. These results can be compared with those obtained in [17]. M. Griebel & F. Kostner's paper [17] uses anisotropic interpolating wavelets with Poisson solver to solve the "merging of three vortices" in velocity-vorticity formulation. With 10,000 degrees of freedom, their results are of lesser quality (ibid. "overestimation of the rotation of the cores of the vortices") than the ones we obtained with 3,500 degrees of freedom. The computational time for the full wavelet code is about four times the Fourier code in the periodic case for which the Fourier code is known to be nearly optimal. But the wavelet method can be adapted to other cases as the inclusion of boundary conditions Wavelet methods fit better complex geometries with penalisation methods, and may be included in an adaptive code, in contrast with the Fourier method.

While only results in dimension two are presented in this paper, the divergence-free wavelet method extends directly to dimension three.

Finally, this type of schemes could be improved in a number of ways: one can make this method partially lagrangian by convecting the small vortices by the large scales of the flows. In the context of divergence-free wavelets, the velocity field is decomposed into a sum of vortices in a stable way: $\|u\|_{L^2}^2 \sim \sum_{(j,k)\in\Lambda} \|\tau_{(j,k)}\|_{L^2}^2$, with the functions $\tau_{(j,k)}$ similar each to other by dilations and translations. These considerations in two dimensions are extended to dimension three by considering two or three different vortex functions instead of one.

Acknowledgements

The authors gratefully acknowledge the University of Ulm and especially the Numerical Analysis Departement where most of the numerical experiments presented in this paper were conducted. The authors also wish to express their gratitude to Kai Bittner, Nicholas Kevlahan and Karsten Urban for fruitful discussions.

References

[1] C.-M. Albukrek, K. Urban, D. Rempfer, and J.-L. Lumley, *Divergence-Free Wavelet Analysis of Turbulent Flows*, J. of Scientific Computing **17**(1): 49-66, 2002.

[2] P. Charton, V. Perrier, *A pseudo-wavelet scheme for the two-dimensional Navier-Stokes equations*, Comp. Appl. Math. **15**(2): 139-160, 1996.

[3] Chorin, A.J., and J.E. Marsden, *A Mathematical Introduction to Fluid Mechanics*, book, 3rd ed., Springer, 1993.

[4] A. Cohen, I. Daubechies and J.C. Feauveau, *Biorthogonal bases of compactly supported wavelets*, Comm. Pure Appl. Math., 45, 485-560, 1992.

[5] A. Cohen, *Wavelet methods in numerical analysis*, Handbook of Numerical Analysis, vol. VII, P.G. Ciarlet and J.L. Lions eds., Elsevier, Amsterdam, 2000.

[6] W. Dahmen, A. Kunoth and K. Urban, *A wavelet-Galerkin method for the Stokes problem*, Computing 56, 259-302, 1996.

[7] E. Deriaz, *Ondelettes pour la Simulation des Écoulements Fluides Incompressibles en Turbulence*, Thèse de doctorat de l'INP Grenoble, 2006.

[8] E. Deriaz, *Shannon wavelet approximations of linear differential operators*, Preprint IMPAN 676, january 2007.

[9] E. Deriaz and V. Perrier, *Divergence-free Wavelets in 2D and 3D, application to the Navier-Stokes equations*, J. of Turbulence, **7**(3): 1–37, 2006.

[10] E. Deriaz, K. Bittner and V. Perrier, *Décomposition de Helmholtz par ondelettes : convergence d'un algorithme itératif* (in french), [Wavelet Helmholtz decomposition: convergence of an iterative algorithm], ESAIM: Proceedings, in press.

[11] E. Deriaz and V. Perrier, *Orthogonal Helmholtz decomposition in arbitrary dimension using divergence-free and curl-free wavelets*, Preprint IMPAN 680 submitted to Appl. Comput. Harmon. Anal.

[12] M. Farge, *Wavelet transforms and their applications to turbulence*, Ann. Rev. Flu. Mech. **24**: 395-457, 1992.

[13] M. Farge, N. Kevlahan, V. Perrier and E. Goirand, *Wavelets and turbulence*, Proc. IEEE 84(4), 639-669, 1996.

[14] M. Farge and K. Schneider, *Coherent Vortex Simulation (CVS), A Semi-Deterministic Turbulence Model Using Wavelets*, Flow, Turbulence and Combution, **66**: 393-426, 2001.

[15] J. Fröhlich and K. Schneider, *Numerical simulation of decaying turbulence in an adaptive wavelet basis*, Appl. Comput. Harmon. Anal., **3**: 393-397, 1996.

[16] V. Girault, P.A. Raviart, *Finite element methods for Navier-Stokes equations*, Springer-Verlag Berlin, 1986.

[17] M. Griebel and F. Koster, *Adaptive wavelet solvers for the unsteady incompressible Navier-Stokes equations*, Advances in Mathematical Fluid Mechanics, J. Malek and J. Necas and M. Rokyta eds, Springer-Verlag, 2000.

[18] S. Grossmann and M. Löhden, *Wavelet analysis of Navier-Stokes flow*, Z.Phys.B, **100**: 137-147, 1996.

[19] J. Ko, A.J. Kurdila and O.K. Rediniotis, *Divergence-free Bases and Multiresolution Methods for Reduced-Order Flow Modeling*, AIAA Journal, **38**(2): 2219-2232, 2000.

[20] F. Koster, M. Griebel, N. Kevlahan, M. Farge and K. Schneider, *Towards an adaptive wavelet-based 3D Navier-Stokes solver*, Numerical flow simulation I, Notes on Numerical Fluid Mechanics, **66**: 339–364, E.H. Hirschel eds, Vieweg-Verlag, Braunschweig, 1998.

[21] J.-P. Kahane and P.G. Lemarié-Rieusset, *Fourier series and wavelets*, book, Gordon & Breach, London, 1995.

[22] N. Kevlahan, O.V. Vasilyev, D. Goldstein and A. Jay, *A three-dimensional adaptive wavelet method for fluid–structure interaction.* In *Direct and Large-Eddy Simulation V*, (ed. B. J. Geurts, R. Friedrich & O. Métais). 8 pp. Kluwer, 2004.

[23] N.K.-R. Kevlahan and O.V. Vasilyev, *An adaptive wavelet collocation method for fluid–structure interaction at high Reynolds numbers.* SIAM J. Sci. Comput. **26**(6), 1894-1915, 2005.

[24] P.G. Lemarié-Rieusset, *Analyses multi-résolutions non orthogonales, commutation entre projecteurs et dérivation et ondelettes vecteurs à divergence nulle* (in french), Revista Matemática Iberoamericana, **8**(2): 221-236, 1992.

[25] J. Lewalle, *Wavelet transform of the Navier-Stokes equations and the generalized dimensions of turbulence*, Appl. Sci. Res. **51**(1-2):109-113, 1993.

[26] J. Liandrat, P. Tchamitchian, *On the fast approximation of some non linear operators in non regular wavelet space*, Advances in computational mathematics **8**: 179-192, 1998.

[27] S. Mallat, *A Wavelet Tour of Signal Processing*, Academic Press, 1998.

[28] C. Meneveau, *Analysis of turbulence in the orthonormal wavelet representation*, Journal of Fluid Mechanics **232**: 469-520, 1991.

[29] P. Monasse and V. Perrier, *Orthonormal wavelet bases adapted for partial differential equations with boundary conditions*, SIAM J. on Math. Analysis **29**(4): 1040-1065, 1998.

[30] K. Schneider, N. Kevlahan and M. Farge, *Comparison of an adaptive wavelet method and nonlinearly filtered pseudo-spectral methods for two-dimensional turbulence*, Theor. Comput. Fluid Dyn. **9**: 191-206, 1997.

[31] K. Schneider and M. Farge, *Numerical simulation of a mixing layer in an adaptive wavelet basis*, C. R. Acad. Sci. Paris, série II, 263-269, 2000.

[32] K. Urban, *A Wavelet-Galerkin Algorithm for the Driven–Cavity–Stokes–Problem in Two Space Dimensions*, RWTH Aachen, Preprint 1994.

[33] K. Urban, *Using divergence-free wavelets for the numerical solution of the Stokes problem*, AMLI'96: Proceedings of the Conference on Algebraic Multilevel Iteration Methods with Applications, **2**: 261–277, University of Nijmegen, The Netherlands, 1996.

[34] K. Urban, *Wavelet Bases in H(div) and H(curl)*, Mathematics of Computation **70**(234): 739-766, 2000.

[35] K. Urban, *Wavelets in Numerical Simulation*, Springer, 2002.

[36] O.V. Vasilyev and N.K.-R Kevlahan, *An adaptive multilevel wavelet collocation method for elliptic problems*, J. Comp. Phys. **206**: 412-431, 2005.