Towards an adaptive wavelet-based 3D Navier-Stokes Solver

F. Koster[1], M. Griebel[1],
N. K.-R. Kevlahan[2], M. Farge[2], K. Schneider[3]

[1] Inst. f. Angew. Math. Universität Bonn, Wegelerstr.6, D-53115 Bonn
[2] LMD-CNRS, Ecole Normale Sup., 24 rue Lhomond, F-75231 Paris cedex 05
[3] CPT-CNRS Luminy, Case 907, F-13288 Marseille cedex 09

Summary

In this paper we present results obtained using wavelet methods for the analysis and simulation of two-dimensional turbulence, and a preliminary study of a three-dimensional turbulent channel flow. The two-dimensional results show the efficiency of wavelets for the numerical simulation of turbulence, and suggest new methods for modelling the flow, based on a decomposition into coherent and incoherent parts. The three-dimensional study compares different wavelet type approaches applied to instantaneous velocity, vorticity and pressure fields. At least for the data and the setting considered in our experiments, i.e. flows with dominating boundary layer, the results indicate which of the various forms of the three-dimensional Navier–Stokes Equations (3D-NSE) (primitive variables, vorticity-streamfunction formulation) and which multiscale approach is well suited for the numerical integration of the 3D-NSE. In addition, we present efficient techniques for the implementation and parallelization of an adaptive wavelet-based 3D-NSE solver.

1 Introduction

In both two and three dimensions turbulence is characterized by localized regions of strong variations in the quantities describing the flow. These coherent structures take the form of roughly circular patches of strong vorticity in two dimensions [25] and thin tubes of strong vorticity in three dimensions [23]. The existence of these tubes has been confirmed many times both in numerical [31], [34] and laboratory experiments [3]. The tubes have diameters of the order of the Kolmogorov scale and lengths up to the integral scale. The interest of these patches or tubes is that they are believed to control the dynamics of the flow. This is the foundation of the various vortex-based methods [24] in both two and three dimensions. Furthermore, the localized nature and sometimes complex internal structure of the vortices suggests the use of multiscale methods. Due to their localization in scale and space these methods are good candidates to reduce the complexity of the flow. One example of multiscale methods, wavelets, have been used for the analysis and simulation of two-dimensional turbulence with periodic boundary conditions [11], [13], [5]. The results in two dimensions have been encouraging and suggest new modelling methods (see §3), and it is clear that the next step is to attempt to apply similar methods to three-dimensional flows and flows with boundaries.

This paper has four main sections. The first part (§2) briefly describes multiscale methods in both two and three dimensions. Starting from three particular univariate approaches we will explain two constructions for multivariate methods. One of these approaches is new in the context of wavelets, but gives superior results in three dimensions.

The second part (§3) describes some new results obtained by the French members of our group for the analysis and simulation of two-dimensional turbulence using wavelets. The goal is to demonstrate that wavelet techniques can be efficient and highly accurate for computing two-dimensional turbulent flows. Furthermore, wavelet analysis shows that the flow may be objectively separated into a coherent part (with non-Gaussian vorticity statistics) and an incoherent part (with Gaussian vorticity statistics). This result suggests a new highly efficient modelling method where the coherent part of the flow is calculated using the wavelet solver, while the incoherent part is modelled statistically or merely advected by the coherent part.

In the third part (§4) we applied different multiscale methods to a database of a 3D turbulent channel flow. This work was done in Bonn. By the comparison of compact representations of the velocity and vorticity we aim to answer the following questions:

- Do we need to solve the 3D-NSE in terms of vorticity or velocity in order to exploit the localization of the coherent structures?
- Which of the various multiscale methods yield a good trade-off between numerical efficiency and suitability for turbulence simulations?
- How can we implement an adaptive method efficiently?

The main, but still preliminary, result of this part is that at least for the flow under consideration the formulation of the 3D-NSE in primitive variables seems to be better suited to the integration than the formulation in terms of vorticity. In part, this is due to the fact that in three dimensions the numerical costs for the vorticity formulation are much higher. The second result is that simple multiscale methods (prewavelets) give a very good trade-off between reduction of the complexity of the flow and numerical efficiency. This analysis will form the foundation for the implementation of a wavelet-based 3D-NSE solver by our group. Some algorithmic details for efficient data management and parallelization of the adaptive solver are discussed in §5. Note that the use of such techniques is crucial for adaptive codes, in order to preserve their advantage of reducing the complexity of the physical problem by working with a sparse representation.

2 Multiscale Methods

We will briefly describe some topics from the theory of multiscale methods. These lines are in the spirit of the standard literature, see also the recent introducing paper of Dahmen [7].

2.1 General Concept of Multiscale Analysis

For a multiscale analysis (MSA) of a class H of functions one considers a sequence of nested subspaces $V_j \subset H$, whose union is dense in H, and complementary spaces $W_j = V_{j+1} \ominus V_j$.

$$\{0\} = V_0 \subset ... \subset V_j \subset V_{j+1} \subset ... H \quad ; \quad W_j \oplus V_j = V_{j+1} \quad \text{and} \quad clos \bigcup_{j=0}^{\infty} V_j = H$$

Let V_j and W_j resp. have bases $\{\phi_l^j\}_{l\in I_j}$ and $\{\psi_l^j\}_{l\in J_j}$, where I_j, J_j are some index sets. For the following explanations it will be convenient, to write the bases as (possibly infinite) column vectors Φ^j and Ψ^j. For the MSA of a given function $u \in H$ one is looking for coefficients $u^j := \{u_l^j\}_{l\in J_j}$ of the expansion

$$u = \sum_{j=0}^{\infty} (u^j)^T \cdot \Psi^j \tag{2.1}$$

With some additional requirements on H and the bases Φ^j and Ψ^j, the coefficients u^j decay with increased *scale* j

$$|u^j|_{l_p} \to 0 \quad , \quad j \to \infty \quad .$$

Further assumptions on H, Φ^j and Ψ^j then will allow for a good approximation of u by an expansion similar to (2.1) using only a very limited number of significant coefficients u_l^j

$$\tilde{u} = \sum_{j,l \;;\; |u_l^j| > \epsilon(j)} u_l^j \psi_l^j \quad \text{and} \quad \|u - \tilde{u}\|_H \quad \text{is small.} \tag{2.2}$$

In this equation $\epsilon(j)$ is a certain cut-off parameter, which here depends only on the scale j, but may be chosen in a more sophisticated manner.
In practical applications usually one has a representation $u = (c^J)^T \cdot \Phi^J \in V^J$, derived from collocation values of u by, e.g. numerical quadrature methods [33], and one is looking for the coefficients $\{u^j\}_{0\leq j\leq J}$. This corresponds to the transform $T^J : c^J \mapsto \bigcup_{j=0}^{J-1} u^j$.
Often, besides the sequence V^j of *trial* spaces, one has explicit knowledge of an ascending sequence of *test* spaces V_j^* and their complements W_j^*, such that the bases Φ^{*j}, Ψ^{*j} for a certain dual pairing $< .,. >$ fulfill *biorthogonality* relations

$$< \Phi^j, \Psi^{*j} >= \mathbf{0} \quad ; \quad < \Psi^j, \Phi^{*j} >= \mathbf{0} \quad ; \quad < \Phi^j, \Phi^{*j} >= I_{I_j} \quad ; \quad < \Psi^j, \Psi^{*j} >= I_{J_j}$$

The nestedness of the spaces $V_j^*, W_j^* \subset V_{j+1}^*$ leads to the existence of *refinement matrices*, such that

$$\Phi^{*j} = \mathcal{H}_j^* \cdot \Phi^{*j+1} \quad ; \quad \Psi^{*j} = \mathcal{G}_j^* \cdot \Phi^{*j+1} \quad .$$

Similar relations hold for the test spaces involving matrices $\mathcal{H}_j$ and $\mathcal{G}_j$. Together with biorthogonality this implies the following recursive scheme (Fast Wavelet Transform)

$$c^J = < u, \Phi_J^* > \quad ; \quad c^j = \mathcal{H}_j^* c^{j+1} \quad \text{and} \quad u^j = \mathcal{G}_j^* c^{j+1} \quad \text{for} \quad 0 \leq j < J \quad .$$

Hence, T^J is formally given by the chain $T^{1,0} \circ .. \circ T^{J,J-1}$ of operators, each $T^{j,j-1}$ mapping the coefficients $\bigcup_{k=j}^{J-1} u^k \bigcup c^j$ to $\bigcup_{k=j-1}^{J-1} u^k \bigcup c^{j-1}$. Whenever the matrices $\mathcal{H}_j^*$ and $\mathcal{G}_j^*$ are sparse, uniformly in j, the transform T^J can be applied very fast. A moments thought yields that the inverse operation $(T^J)^{-1}$ is given by the recursive scheme $c^{j+1} = (\mathcal{H}_j)^T c^j + (\mathcal{G}_j)^T u^j \quad (0 \leq j < J)$.

2.2 Univariate Examples

We will briefly introduce three families of univariate biorthogonal systems. From these we build up the multivariate systems used in our experiments. The three families have in common, that they are generated by dilates and translates of 'mother' functions $\phi^{(*)}$, $\psi^{(*)}$. The families are the hierarchical nodal basis, pre-wavelets and an example for compactly supported orthogonal wavelets. Figure 1 shows some of the trial functions.

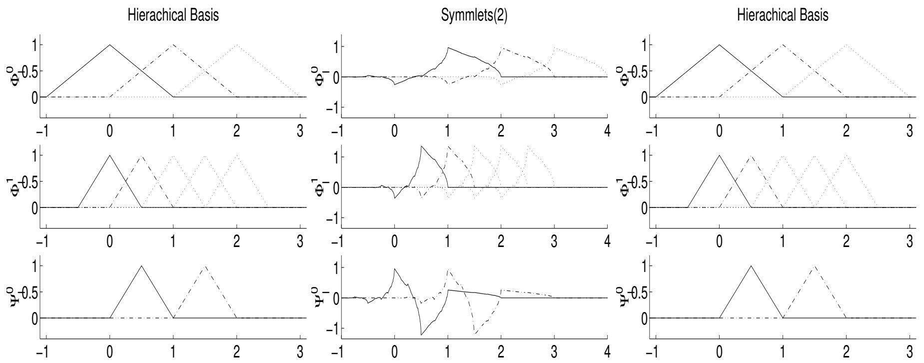

Figure 1: Univariate trial functions

(HB) Hierarchical Nodal Basis for $H = L_2([0,1])$:
Let $\phi(x) = \max\{0, 1 - |x|\}$ denote the well-known 'hat' function. Then, suitable basis functions for V_j and V_j^* are

$$\phi_l^j := \phi(2^j x - l) \quad \text{and} \quad \phi_l^{*j} := \delta(2^j x - l) \quad ; \quad 0 \leq l \leq 2^j).$$

Here δ denotes the Dirac functional. Bases of the complementary spaces are given by

$$\psi_l^{j-1} = \phi(2^j x - 2l + 1) \quad \text{and} \quad \psi_l^{*j-1} = \phi_{2l-1}^{*j} - \frac{1}{2}(\phi_{2l-2}^{*j} + \phi_{2l}^{*j}) \quad ; \quad 1 \leq l \leq 2^j - 1.$$

(OW) Compactly Supported Orthogonal Wavelets for $H = L_2(\mathbf{R})$:
In this case ϕ is the solution of the equation

$$\phi(x) = \sqrt{2} \sum_{l=-L}^{R} h_l \phi(2x - l) \ ,$$

with some very distinct coefficients h_l. Solutions of such equations and the corresponding coefficients were first given by Daubechies [8], [9]. The L_2-orthogonal basis functions are the dilates and translates $\phi_l^j = \phi_l^{*j} = 2^{j/2}\phi(2^j x - l)$ and $\psi_l^j = \psi_l^{*j} = 2^{j/2}\psi(2^j x - l)$ for $l \in \mathbf{Z}$, where $\psi(x) = \sqrt{2}\sum_{l=-R+1}^{L+1}(-1)^l h_{1-l}\phi(2x - l)$.

In numerical applications one usually deals with bounded domains, intervals in the simplest case, thus the framework for $L_2(\mathbf{R})$ above is not satisfactory and one has to modify at least the basis functions ϕ_l^j, ψ_l^j near the boundaries. Such a construction is given by [15], generalizing and improving previous results of [6], [27] with respect to the numerical stability of the boundary adaptation. In our experiments we used boundary adapted least asymmetric wavelets of the Daubechies family with two vanishing moments (Symmlets(2)).

(PRW) Lifting Prewavelets:
A simple, but very powerful, method for the construction of biorthogonal systems has been given by W. Sweldens [32]. In terms of the refinement matrices biorthogonality reads

$$\begin{pmatrix} \mathcal{H}_j \\ \mathcal{G}_j \end{pmatrix}^T \cdot \begin{pmatrix} \mathcal{H}_j^* \\ \mathcal{G}_j^* \end{pmatrix} = \begin{pmatrix} I & \mathbf{0} \\ \mathbf{0} & I \end{pmatrix} \quad . \tag{2.3}$$

These relations are preserved, if we switch to a biorthogonal system with refinement matrices

$$\begin{pmatrix} \tilde{\mathcal{H}}_j \\ \tilde{\mathcal{G}}_j \end{pmatrix} = \begin{pmatrix} I & \mathbf{0} \\ K & I \end{pmatrix} \cdot \begin{pmatrix} \mathcal{H}_j^* \\ \mathcal{G}_j^* \end{pmatrix} \quad ; \quad \begin{pmatrix} \tilde{\mathcal{H}}_j^* \\ \tilde{\mathcal{G}}_j^* \end{pmatrix} = \begin{pmatrix} I & -K^T \\ \mathbf{0} & I \end{pmatrix} \cdot \begin{pmatrix} \mathcal{H}_j^* \\ \mathcal{G}_j^* \end{pmatrix} \quad .$$

This allows for the modification of existing biorthogonal systems to custom design a new one with special desired features. In our case we choose the primal system corresponding to the hierarchical nodal basis (HB) and K such that the new system exhibits the following properties,

$$< P, \tilde{\Psi}^j > = < P, \tilde{\Psi}^{*j} > = 0 \qquad \text{for each polynomial of degree less than 2}$$

The second condition was also valid for the hierarchical nodal basis, but the first was not. The purpose of this modification comes from the wish for easy preconditioning the Laplace or Helmholtz problems, which have to be solved as part of the integration of the 3D-NSE.

Some properties of these biorthogonal systems are compared in Table 1.

	relative num. work count	preconditioning in 3D
HB	1	No
PRW	2	Yes
OW	5	Yes

Table 1: Work count/preconditioning for biorth. bases

2.3 Multivariate Constructions

In the experiments of the next section, H is a space of functions over the domain $[x_1, x_2] \times [y_1, y_2] \times [z_1, z_2]$. This tensor product structure allows for the following two multivariate approaches. We restrict the presentation to the two-dimensional case, since the generalization to higher dimensions is straightforward.

(ISO) Isotropic approach:
For this construction the bivariate Spaces $\mathbf{V}_j$ are defined by

$$\mathbf{V}_j = V_j \otimes V_j \quad ; \quad \mathbf{W}_j = (V_{j-1} \otimes W_{j-1}) \oplus (W_{j-1} \otimes W_{j-1}) \oplus (W_{j-1} \otimes V_{j-1})$$

The transform formally is given by $\mathbf{T}^J = \bigcirc_{j=1}^{J-1} (T^{j,j-1} \otimes T^{j,j-1})$.
This means we have to apply at each stage of the transform first the univariate $T^{j,j-1}$ with respect to x for all y, leading to the decomposition $\mathbf{V}_j = (W_{j-1} \otimes V_j) \oplus (V_{j-1} \otimes V_j)$, and then with respect to y for all columns, leading to the final decomposition $\mathbf{V}_j = \mathbf{V}_{j-1} \oplus \mathbf{W}_{j-1}$.
Since the univariate spaces V_j, W_j are usually spanned by equally dilated basis functions, the basis of $\mathbf{W}_j$ consists of functions with equal dilation in both coordinate directions. Simply stated, they are isotropic. This approach is the classical one in the framework of multivariate MSA.

(ANISO) Anisotropic and isotropic basis functions:
In contrast to the ISO approach the ANISO construction works in the same way as the multivariate FFT. First we apply the complete decomposition with respect to x and then with respect to y. Formally this leads to the transform $\mathbf{T}^J = T^J \otimes T^J$. Despite of its easy application, this construction leads to a more complicated hierarchy of subspaces, which now depend on two parameters.

$$\begin{aligned} \mathbf{V}_{j,J} &= V_j \otimes V_j \oplus_{k=j+1}^{J} (W_k \otimes V_j) \oplus (V_j \otimes W_k) \\ \mathbf{W}_{j,J} &= \oplus_{k=j}^{J} (W_k \otimes W_j) \oplus (W_j \otimes W_k) \end{aligned}$$

Further information about such multiparametric space splittings are given in [16], [28]. This approach exchanges the 3 types (corresponding $V_{j-1} \otimes W_{j-1}$, $W_{j-1} \otimes W_{j-1}$, $W_{j-1} \otimes V_{j-1}$) of equally dilated basis functions of the complementary spaces with a larger number of different types of basis functions, each having a characteristic dilation in the first and second coordinate direction. Thus there are both, isotropic and anisotropic basis functions. In the experiments of the fourth section it turns out, that this approach is superior compared to ISO, although for fixed J and fixed underlying univariate MSA the numerical cost is larger by a factor of $\frac{7}{4}$ in the three-dimensional case.

2.4 Error Estimation

A simple estimate for the error is

$$||u - \sum_{j,l;\ |u_l^j| > \epsilon(j)} u_l^j \psi_l^j ||_H \leq \sum_{j,l;\ |u_l^j| \leq \epsilon(j)} |u_l^j| ||\psi_l^j||_H \tag{2.4}$$

For our examples, except near the boundaries, the basis functions ψ_l^j are translates of some $\psi_{l_0}^j$. Hence, for inner functions ψ_l^j the norm $||\psi_l^j||_H$ does not depend on l. Together with the estimate above this suggests to use the estimators

(L2) $\qquad \epsilon(j) = \epsilon / ||\psi_{l_0}^j||_{L_2}$

(H1) $\qquad \epsilon(j) = \epsilon / ||\psi_{l_0}^j||_{H_1}$

Here ϵ is a predefined cut-off value.

For the ANISO construction there is another simple a priori error estimator. This estimator essentially recovers the idea of so called sparse grids [35], [4]. The three underlying univariate basis functions span the space of polynomials of degree less than 2. This implies that for functions u with bounded mixed derivatives $||\frac{\partial^4 u}{\partial^2 x \partial^2 y}||_H$, the following estimate is appropriate

$$|u_l^j| ||\psi_l^j||_{L_2} \quad \leq \quad a^{j_1 + j_2} \quad ; \qquad \text{for some } a < 1$$

This suggests setting

(SPG) $$\epsilon(j) = \begin{cases} 0 & ; \quad j_1 + j_2 \leq L \\ \infty & ; \quad \text{else} \end{cases}$$

for an integer cut-off parameter $L \leq 2J$, which controls the compression rate. A very similar technique, which essentially uses the same kind of adaptivity in scale, but in a slightly different manner, has been successfully applied to the simulation of a turbulent pipe flow [17].

At the present point some remarks are in order. The three univariate approaches, introduced in this paper, were built by dilation and translation. This gives rise to some problems when the data is given by collocation values on nonuniform grids, e.g. this is the case for the database used in the fourth section. Of course one could deal with this problem by modifying the quadrature $c^J = < u, \Phi^J >$, but this is rather complicated. Thus we use a much cheaper solution. In a curvilinear NSE solver the data $(\mathbf{u}, p, ...)$ is given on an equidistant, rectangular grid. We treat the data in the same manner and hence, can apply the fast transforms for equidistant grids. However, we must change the error estimator.

Consider data $\{u_i\}$ given on the non-equidistant grid $a = x_1 < ... < x_n = b$. To consider this data as given on an equidistant grid means to identify $\{u_i\}$ with collocation values of $u \circ x(\xi)$, where $x : [0,1] \to [a,b]$; $\xi \mapsto x(\xi)$ is the gradiation function, which satisfies $x' := \frac{\partial x}{\partial \xi} > 0$. Applying the transform yields

$$u \circ x(\xi) = \sum_{j,l} u_l^j \psi_l^j(\xi) = \sum_{j,l} u_l^j \tilde{\psi}_l^j \circ x(\xi) \quad ,$$

with some implicitly defined basis functions $\tilde{\psi}_l^j$ on $[a,b]$. Clearly we should use $\|\tilde{\psi}_l^j\|_{H([a,b])}$ as weights in (L2) or (H1). Especially for the later we obtain

$$|\tilde{\psi}_l^j|^2_{H_1([a,b])} = \int_a^b \frac{|\psi_l^j \circ x(\xi)|^2}{x'^2(\xi)} dx = \int_0^1 \frac{|\psi_l^j \circ x(\xi)|^2}{x'(\xi)} d\xi = \frac{1}{x'(\xi_l^j)} |\psi_l^j|^2_{H_1([0,1])}$$

for some $\xi_l^j \in supp\ \psi_l^j$. In regions of strong gradiation x' is very small, hence $\epsilon(j,l)$ will be small. This is exactly what one would expect. Note, in contrary to modifying the quadrature, this approach is easy to extend to arbitrary curvilinear transformed domains and thus is well suited for use in *adaptive curvilinear* codes.

The next issue concerns orthogonal wavelets. For these one has $\|\psi_l^j\|_{L_2} = 1$ and equality in the estimate (2.4), which is optimal for the non linear thresholding. However, application of the orthogonal wavelet transform to $u \circ x$ can be reinterpreted as application of a non orthogonal transform, i.e. Parseval's identity $\|u\|_0^2 = \sum_{j,l} |u_l^j|^2$ does not hold anymore. Instead of that there are some constants $c(j)$, $C(j)$, such that

$$c(j_0) \sum_{l,\ j \geq j_0} |u_l^j| \|\psi_l^j\|^2 \leq \| \sum_{l,\ j \geq j_0} u_l^j \psi_l^j \|^2 \leq C(j_0) \sum_{l,\ j \geq j_0} |u_l^j| \|\psi_l^j\|^2$$

The condition number $\max_j \frac{C(j)}{c(j)}$ gives the deviation from L_2-orthogonality. The point is that the constants behave like

$$c(j) = 1 - \mathcal{O}(2^{-j}) \qquad ; \qquad C(j) = 1 + \mathcal{O}(2^{-j}) \quad \text{, i.e.}$$

for fine scales (large j), where most of the compression happens, the nice features of orthogonal wavelets are almost recovered by using the weighted cut-off values.

3 Two-dimensional isotropic turbulence

3.1 Goals and methods

In this section we present results obtained using wavelets to analyze and calculate two-dimensional isotropic turbulence with periodic boundary conditions. These results make use of the 2D-NSE solver developed by Fröhlich and Schneider [13] and build on the wavelet analysis techniques developed by Farge *et al.* [11]. The goals of the present work were to verify the accuracy and efficiency of the wavelet based 2D-NSE solver against the usual spectral methods, to develop new wavelet-based forcing methods that avoid the problems inherent in the traditional methods, to develop an objective criterion for separating the coherent and incoherent parts of the flow (based on the different statistical properties of the two parts), and to show that the incoherent part of the flow is stable near the coherent vortices.

3.2 Wavelet simulation

The wavelet transform was first introduced as an analysis technique, but numerical methods have been developed recently which use wavelet bases to actually solve partial differential equations [11], [13], [5]. These methods are particularly well-suited to equations, such as the Navier–Stokes equations at high Reynolds number, whose solutions contain isolated multi-scale structures or quasi-singularities. We compared simulations using these wavelet techniques with standard spectral simulations and nonlinearly filtered spectral simulations [30]. The evolution of the vorticity field for each of the four simulation methods is shown in Figure 2. Comparing the four simulations, one notices that the nonlinear Fourier filtering deforms the edge of the vortices and produces spurious oscillations that quickly spread to fill the whole background flow. The error in representing the edges of the vortices is due to the poor ability of Fourier methods to represent localized sharp gradients; this is exacerbated by the filtering. The homogeneous distribution of the error is not surprising since each Fourier mode is completely de-localized in physical space and thus any error is immediately spread over all space.

The results showed that the wavelet methods are very accurate, and require roughly four times fewer active modes than spectral methods. Furthermore, the number of active wavelet modes is approximately constant in time, even during intense nonlinear interactions, whereas the number of active spectral modes peaks when the interactions are most intense (see Figure 3). It is important to note that in this investigation the non-active modes were simply thrown away. If the modes were modelled in some way, the number of wavelet modes actually calculated would be much smaller (of the order of 1%) and thus the wavelet method would become even more efficient. This possibility is discussed below.

3.3 Wavelet forcing

In order to obtain a statistically steady state in a pseudo-spectral simulation the usual method is to force at a few intermediate scale modes. This method is based on the idea (derived from statistical theories of turbulence) that energy is injected at a particular length-scale (or wavenumber) and then (in two dimensions) cascades to larger scales. The problem with this technique is that it applies a statistical idea to the calculation of an individual flow realization. Physically, it is not reasonable to force by injecting energy at a particular length-scale since we know that turbulent flows are actually forced by the production of vorticity by instability at boundaries. This vorticity detaches from the boundary and is advected into the interior of the flow. In practice, forcing at a single wavenumber does not even produce approximately constant total energy and enstrophy: the

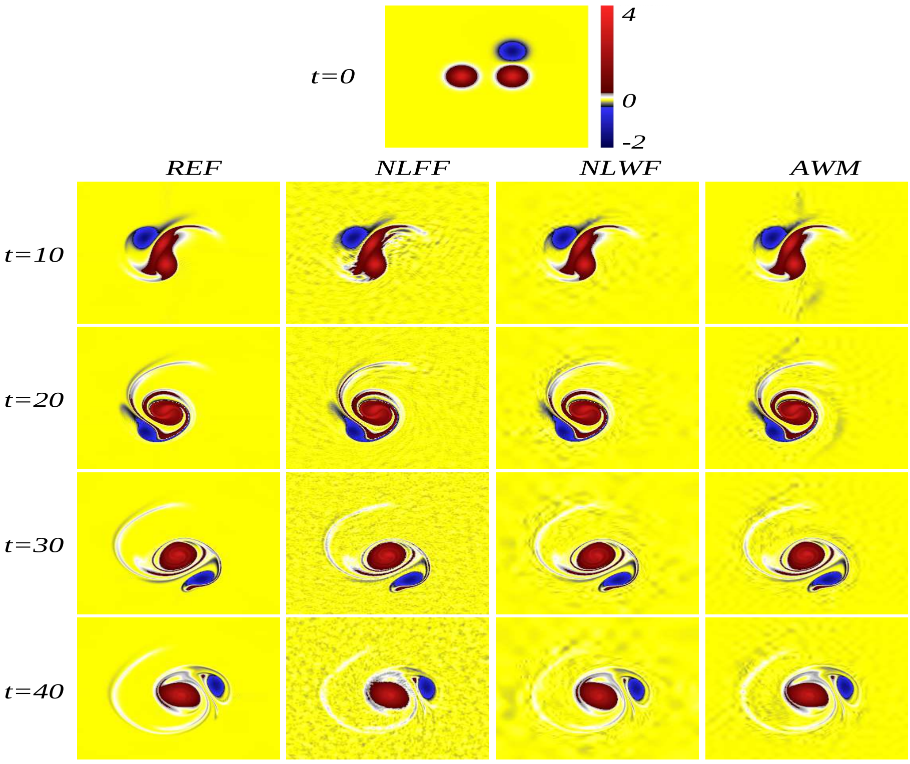

Figure 2: Evolution of the vorticity field as a function of time for each of the four methods. (REF = pseudo-spectral reference simulation, NLFF = nonlinear Fourier filtering, NLWF = nonlinear wavelet filtering, AWM = adaptive wavelet method).

energy and enstrophy fluctuate significantly about a mean. If this mean does not change the flow is said to be statistically stationary. Clearly, this is a rather weak definition of stationarity.

The large wavelet modes correspond to the coherent vortices of the flow. Based on this observation we have introduced a new forcing method that directly amplifies the vortices of the flow by reinforcing the large wavelet modes [29]. This forcing method injects energy and enstrophy into the vortices (rather than into a wavenumber) and thus corresponds more closely to the physical forcing of the turbulent flows than the usual method described above. By forcing in wavelet space we can also control the smoothness of the excited vortices. Wavelet forcing was shown to produce almost constant total enstrophy and energy; a significant improvement on the wavenumber-based method. We also checked that the probability distribution functions (PDF) of vorticity and the energy spectrum do not evolve. This new forcing method should allow the simulation of more realistic

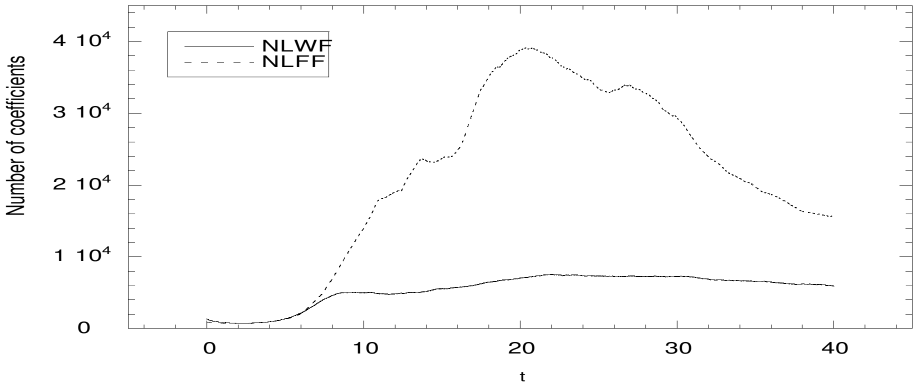

Figure 3: Evolution of the number of active modes for each method (NLFF = nonlinear Fourier filtering, NLWF = nonlinear wavelet filtering and adaptive wavelet simulation).

stationary isotropic two-dimensional turbulent flows, even with periodic boundary conditions.

3.4 Coherent structure eduction

In order to develop effective vortex-based methods for solving or modelling the 2D-NSE it is vital to have an objective and efficient criterion for separating the coherent vortices from the incoherent part of the flow. The separation should ideally ensure that the incoherent part has well-defined simple statistical properties so that it can be easily modelled or calculated.

Recently, Donoho [10] developed a de-noising technique to remove Gaussian white noise from a signal. He showed that if one applies the following threshold $\widetilde{\omega_T}$ to the wavelet modes

$$\widetilde{\omega_T} = (2\log_{10}(N))^{1/2}\sigma/N^{1/2} \tag{3.5}$$

where N is the number of points, and σ is the variance of the noise, then one can extract the signal from the Gaussian background noise. We decided to turn Donoho's theory around and *suppose* that the turbulent vorticity contains a Gaussian component whose variance is simply given by the total variance of the vorticity [12]. Note that the threshold should be constant for a stationary flow.

We have analyzed vorticity fields obtained from direct numerical simulations (DNS) of statistically stationary two-dimensional turbulence where the forcing is done in wavelet space. Using the nonlinear wavelet technique based on an objective universal threshold we separate the vorticity field into coherent structures and background flow. Both components are multi-scale with different scaling laws, and therefore cannot be separated by Fourier filtering. We find that the coherent structures have non-Gaussian statistics (and represent only 1.3%

of the total 128^2 modes) while the background flow is Gaussian (and makes up 98.7% of the modes). This result has important implications for modelling two-dimensional turbulence since it shows that the flow can be divided into a component with Gaussian statistics (which can be easily modelled by an equivalent stochastic process), and a component with non-Gaussian statistics (which will have to be calculated exactly). The fact that the non-Gaussian part can be represented with only a small number of modes means that this decomposition should form the basis for a new highly efficient numerical simulation method for two-dimensional turbulence. The interpretation of the weak wavelet modes as the incoherent part of the flow was reinforced by noting that the coherence function (scatter plot of the vorticity versus the streamfunction) is isotropic for the weak wavelet modes, but has a characteristic $sinh$ profile for the strong modes corresponding to the coherent structures.

3.5 Stability of the background flow

The question of the stability of the background flow is important for any vortex method that calculates exactly the evolution of the coherent vortices, but only models the incoherent (or background) part of the flow. If the incoherent part is unstable it generates new vortices, and this process would correspond to a significant forcing of the coherent part by the background flow. Clearly, such an interaction would have to be taken into account in any vortex-based model. Even if the background remains stable it could still affect the evolution of the coherent vortices (e.g. by shielding them from the effect of other vortices), and if this is the case the effect would also have to be included in the model.

We have investigated the stability and effect of the vorticity filaments that are typical of the background flow [21]. To address these questions we used a high-resolution pseudo-spectral DNS to study a simplified model of two-dimensional turbulence: the merging of two vortices accelerated by a third. Vortex merging is the fundamental interaction of two-dimensional turbulence and is also the interaction that generates vorticity filaments. The calculation was carried out on the CRAY C98 of IDRIS (Institut du Développement et des Ressources en Informatique Scientifique) of the CNRS. A wavelet analysis permitted us to cleanly separate the vorticity filaments from the coherent vortices. By comparing the evolution of the filaments with and without the coherent vortices we showed that the filaments are stabilized by the vortices (this comparison is shown in Figure 4). The stabilizing role of the coherent vortices had been suggested on the basis of analytical studies, but had not been verified under realistic conditions. A linear stability analysis and calculation of strain rates then revealed precisely how the stabilization takes place. Finally, by using an asymptotic analysis checked by a DNS we established that the vorticity filaments can protect the vortex they surround from the deforming effects of the weak strain produced by neighbouring vortices.

These results suggest that, provided the density of coherent vortices is sufficient, the background flow should not generate any new coherent vortices. Thus

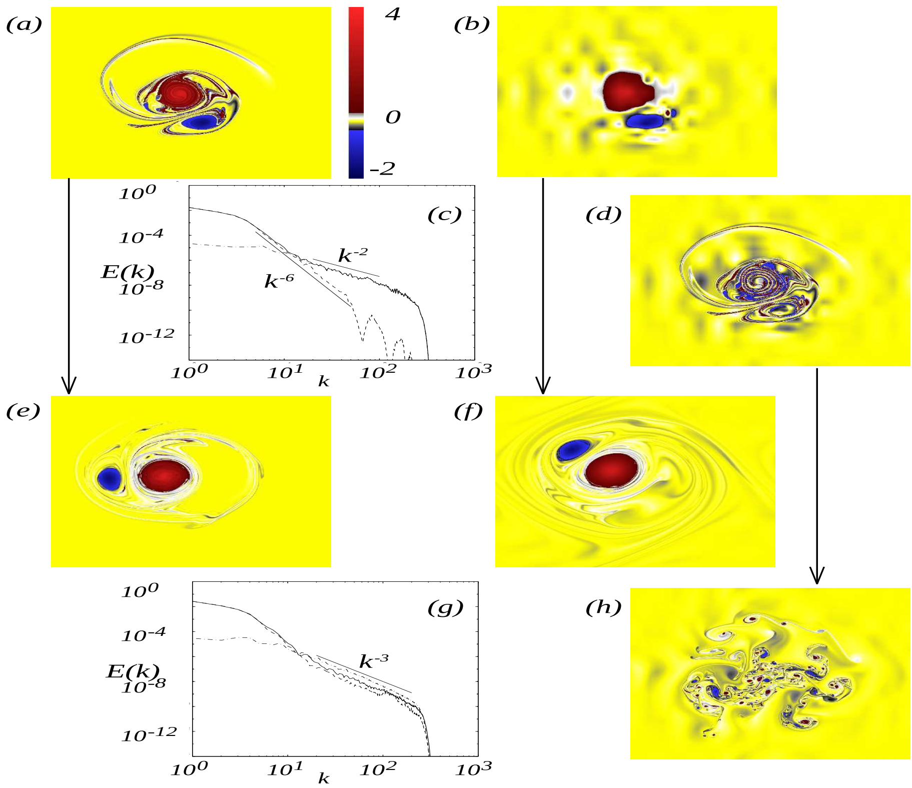

Figure 4: Dynamical analysis of coherent structures and incoherent background flow. (a) Total vorticity at $t = 10$ computed with a resolution 1 024^2. (b) Vorticity corresponding to the coherent vortices alone at $t = 10$. They are made up of 31 strong wavelet packet coefficients which contain 83% of the total enstrophy. (c) Energy spectra at $t = 10$: —, the total energy spectrum; – – –, the coherent vortices energy spectrum; – · –, the filament energy spectrum. (d) Vorticity corresponding to the filaments alone at $t = 10$. They are made up of 1 048 545 weak wavelet packet coefficients which contain 17% of the total enstrophy. (e) Integration of the total vorticity until $t = 30$. (f) Integration of the coherent vortices alone until $t = 30$. (g) Energy spectra at $t = 30$: —, the total energy spectrum; – – –, the coherent vortices energy spectrum; – · –, the filament energy spectrum. (h) Integration of the filaments alone until $t = 30$.

the production of coherent vortices by the background should usually be negligible in any vortex-based model. However, we saw that the background can have an effect on the coherent vortices by reducing their deformation by distant vortices. This shielding effect may reduce the minimum distance for merger and thus change the dynamics of coherent vortex evolution. It remains to be seen whether this effect is important enough that it needs to be included in vortex-based models.

4 Multiscale Analysis of a 3D Turbulent Channel Flow

In order to figure out the potential of multiscale methods for the numerical solution of the 3D Navier Stokes equations, we applied the different methods of the second section to some instantaneous fields of velocity components, pressure and derived quantities of a 3D turbulent channel flow. We choose this flow, because of the simple geometry of the domain and the presence of very different regimes in the flow. It is nearly isotropic in the central flow and strongly anisotropic in the vicinity of the wall. Due to this different behaviour, in general, one uses non equidistant grids for the simulation. For the multiscale methods we used, this leads to a lack of L_2-orthogonality, as described at the end of subsection 2.4. The different flow regimes are representative for most flow configurations with boundaries. Thus, is should be possible to 'extrapolate' the results reported here to other cases of practical interest. However, the case of isotropic turbulence, where equidistant grids are appropriate and allow for L_2-orthogonal wavelets is beyond the scope of the present findings.

4.1 Numerical Experiments

The database we used for our experiments was computed with a second order finite volume code similar to [1] by H.J. Kaltenbach [20], who kindly allowed us to analyze his database. This DNS was run with the usual geometry of upper and

geometry	grid	Δt $[\frac{H}{U_{bulk}}]$	Re_{bulk}	Re_{τ}	u_{τ}
$10h \times 2h \times 4h$	$192 \times 96 \times 192$	0.025	2801	178	0.063546

Table 2: Parameter of DNS by Kaltenbach

lower walls separated with a distance of $2h$, and periodic boundary conditions in stream- and spanwise directions. Characteristical quantities of this DNS are given in Table 2. Figure 5 depicts the different flow regimes: long stretched vortices in the vicinity of the wall and a rather isotropic behavior in the central part of the channel. To this database we applied the following procedure:

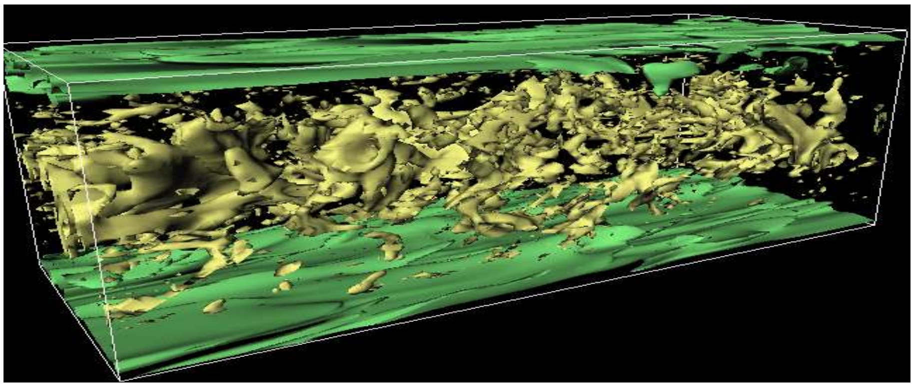

Figure 5: Isosurfaces of modulus of vorticity

transform of $X \in \{u_1, u_2, u_3, \omega_1, \omega_2, .., p, \Delta p\}$ to obtain coefficients X_l^j with respect to the multiscale basis
threshold the significant coefficients, i.e. $\lvert X_l^j \rvert \geq \epsilon(j)$ in (2.2), with respect to one of the criteria L2/H1/SPG and for different rates of compression
inverse transform of the compressed coefficients to obtain $\tilde{X}$
comparison of X and $\tilde{X}$

(Experiment I) In a first series of experiments we run the above procedure for the various quantities. For the comparison in the fourth step we considered mean and r.m.s values of X, $\tilde{X}$ and the error $X - \tilde{X}$.

In addition to these experiments we run a modified type of experiment, where we handle and compress the velocity or vorticity components simultaneously, i.e. we apply the following procedure:

transform of $\mathbf{X} \in \{\mathbf{u} = (u_1, u_2, u_3),\ \mathit{rot}\ \mathbf{u}\}$ to obtain coefficients $\mathbf{X}_{(1)}{}_l^j, .., \mathbf{X}_{(3)}{}_l^j$ with respect to the multiscale basis
threshold in each component $\mathbf{X}_{(1)}{}_l^j, .., \mathbf{X}_{(3)}{}^j$ simultaneously the important coefficients, i.e. $\sqrt{\lvert \mathbf{X}_{(1)}{}_l^j \rvert^2 + .. + \lvert \mathbf{X}_{(3)}{}_l^j \rvert^2} \geq \epsilon(j)$ in the 3-component analogue of (2.2), with respect to one of the criteria L2/H1/SPG and for different rates of compression
inverse transform of the compressed coefficients to obtain $\tilde{\mathbf{X}}$
comparison of $\mathbf{X}$ and $\tilde{\mathbf{X}}$

(Experiment II) Now, we applied the procedure for the three velocity components. Then, for the comparison we took into account not only mean and r.m.s values of $\mathbf{u}$ and $\tilde{\mathbf{u}}$ but also mean and r.m.s values of the derived quantities $\omega = rot\ \mathbf{u}$ and $rot\ \tilde{\mathbf{u}}$.

(Experiment III) Then, we run the above procedure for the three components of the vorticity. For the comparison in the fourth step we considered mean and r.m.s values of $rot\ \mathbf{u}$ and $\widetilde{rot}\ \mathbf{u}$.

From these experiments we will identify multiscale transforms with a good trade-off between work count and compression properties and good criteria for adaptivity. In addition we will see, whether a direct compression of ω (experiment III) or single components of it (experiment I) allows for much higher compression rates (CR) than the evaluation of the vorticity from a compressed velocity field (experiment II). At least for the flow configuration under consideration this tells, whether a formulation of the NSE in primitive variables $\mathbf{u}, p$ is well suited for numerical algorithms and how much we loose by a simultaneous compression of the three components of $\mathbf{u}$ or ω.

In the following the compression rate (CR) denotes

$$\mathrm{CR} = \frac{\text{number of all coefficients}}{\text{number of retained coefficients}},$$

i.e. CR=50 means that only 2% of the coefficients are significant.

4.2 Results of Multiscale Experiments

Before we go into detail, some general remarks are in order. For all multiscale methods we could observe the following effect. Comparing mean and r.m.s values of the error $e := X - \tilde{X}$ with mean and r.m.s values of the original quantity $mean(X)$, $rms(X)$, it turns out that the usual L_2-norm of the error

$$||e||_0^2 = \int e^2 dxdydz \quad ,$$

is of the same order of magnitude than the total turbulent energy of the flow. This seems to indicate that we can *not* expect a good agreement of $rms(\tilde{X})$ and $rms(X)$. But, in fact the contrary is the case. Even for quite large compression rates CR ≥ 50, we find good agreement of $rms(\tilde{X})$ and $rms(X)$.

$$||X - \tilde{X}||_0 = \mathcal{O}(||X - mean(X)||_0) \quad \text{but} \quad rms(\tilde{X}) \cong rms(X)$$

The analogue observations were also obtained for the 3-component fields, i.e. for $\mathbf{X}$ and $\tilde{\mathbf{X}}$. An example for this behavior is given in Figures 6, where r.m.s. values of the streamwise velocity component are compared with those of the compressed data and the compression error. As usual the r.m.s values are scaled by the shear velocity u_τ and the wall distance y is scaled by ν/u_τ. From this observation we conclude that L_2-norms of the error $X - \tilde{X}$ may be misleading, to judge from the quality of the approximation $\tilde{X}$. For a compression rate of CR=63 we obtain in Figure 8 a very good coincidence of e.g $rms(u_2)$ and $rms(\tilde{u}_2)$, although the relative error $\frac{||u_2 - \tilde{u}_2||_0}{||u_2||_0}$ in this case is 31 %, which normally is absolutely unacceptable.

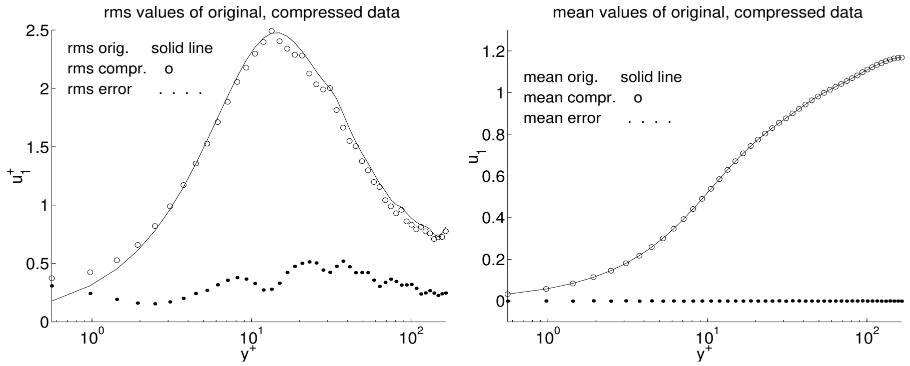

Figure 6: Exp. I /ANISO/PRW/L2 ; CR=48

This phenomenon is due to the property of multiscale basis functions to represent with only a small number of significant coefficients most of the coherent structures in the flow, which seem to govern these statistical quantities (mean, r.m.s). But this is good news, since then the property of compact representation of these important parts should be preserved during the temporal evolution governed by the NSE. However, only experiments with an adaptive NSE solver can give the right answer.

Now we present some more detailed results of our experiments. The most important single parameter for the multiscale transforms was the choice of the multivariate approach ISO or ANISO. As a main result, for all univariate approaches the second construction yielded much better results. One example is given in Figures 7 and 8, which compare the different behavior for prewavelets and the H1 error estimator. For a quite low compression rate of CR=20 the comparison shows, that the isotropic construction ISO is not able to capture the dynamic structures (especially $\overline{u_1' u_2'}$) in the buffer and log-law layer, whereas the second construction ANISO gives very good results even for CR=63.
The analogue result was obtained for the experiment I setting. Of course one might argue, that this is due to the flow regime, with its streaky structures near the wall, which is disadvantageous for the ISO approach. But also in the more central region $y^+ = 40$ to $y^+ = 110$ the r.m.s values of the streamwise velocity component are not recovered. This indicates that the ISO approach performs poorly for physical flows. Hence, in the following comparisons we will only consider results obtained with the ANISO construction.

Another important parameter is the correct error estimation. For modified Symmlets (see the remarks at the end of subsection 2.4) L2 resulted in better approximations, by means of mean and r.m.s. values, than H1. However, for the hierarchical nodal basis and prewavelets in experiments I and III the results for L2 and H1 were similar, while in experiment II the criterion H1 was the best. For both criteria the quality of approximation decreases near the wall. This effect of course is more pronounced for r.m.s values of the vorticity in experiment II (right

Figure 8). Despite of these detractions, mean values are captured satisfactorily (Figure 9). The sparse grid criterion SPG performs not such well. This is the case especially in the buffer and log-law layer. In this region rather coarse and smooth structures generated close to the wall decay to much smaller vortices (Figure 5), which result in significant coefficients in scales, which are simply cut off.

If the multivariate construction is chosen, the main contribution to the numerical costs comes from the underlying univariate approach. Our compression experiments clearly show, that prewavelets perform much better than the hierarchical basis. The additional property of cheap preconditioning Laplace and Helmholtz equations makes them a very promising multiscale technique. Orthogonal wavelets on one hand performed very well in experiments I and III, but in experiment II we could observe significant artifacts near the wall - despite a careful boundary treatment of the transform. If we could solve these problems it might be possible that orthogonal wavelets become an alternative to prewavelets. Experiments, we made with orthogonal wavelets of a higher number of vanishing moments and ignoring the boundary artifacts, seem to indicate this for moderate numbers (2...4) of vanishing moments. At the present state orthogonal wavelets could be applied to the simulation of 3D isotropic turbulence, where equidistant grids allow to completely preserve the advantage of L_2-orthogonality and where boundary artifacts should not be present.

We finish this section with the comparison of the compressed vorticity $\widetilde{rot\,\mathbf{u}}$ and the vorticity of a compressed velocity field $rot\ \tilde{\mathbf{u}}$. As expected the approximation of $rot\ \mathbf{u}$ is somewhat better for $\widetilde{rot\,\mathbf{u}}$ than for $rot\ \tilde{\mathbf{u}}$ (Figures 8, 9 and 12). In particular this holds for r.m.s values immediately near the wall. But the results there are not so much better, to make the vorticity formulation of the NSE competitive to the one in primitive variables, at least from the point of numerical costs. A comparison of the results of experiment I and III, e.g. for ω_3, shows that the simultaneous treatment of all three components is reasonably and does not diminish accuracy (Figures 12, 13).

5 Algorithmical Aspects

From the previous experiments we learned that a relatively small number of coefficients X_l^j (with respect to the multiscale basis and the given quantity $X \in \{\mathbf{u}, p\}$) is sufficient for a reasonable good approximation of X. Thus it is an appealing idea to exploit this for the fast approximative solution of the NSE. Such a spatially adaptive NSE solver computes the approximations $\mathbf{u}_\Lambda$, p_Λ of $\mathbf{u}$, p using only the coefficients $\{(\mathbf{u}, p)_l^j \mid (j,l) \in \Lambda\}$. Here, Λ denotes the set indices (j,l) with significant coefficients X_l^j, which fulfill one of the criteria L2/H1. The basic steps of this approach are given in Table 3.

The locality in space or scale of coherent structures, e.g. vortex tubes, induces the locality of *active* coefficients. This is caused by the multiresolution approach, where $X_l^j \psi_l^j$ represents the difference of X from the present level to the next

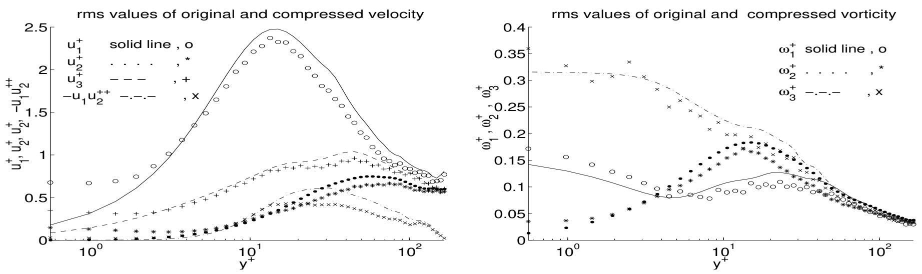

Figure 7: Exp. II/ISO/PRW/H1 ; CR=20

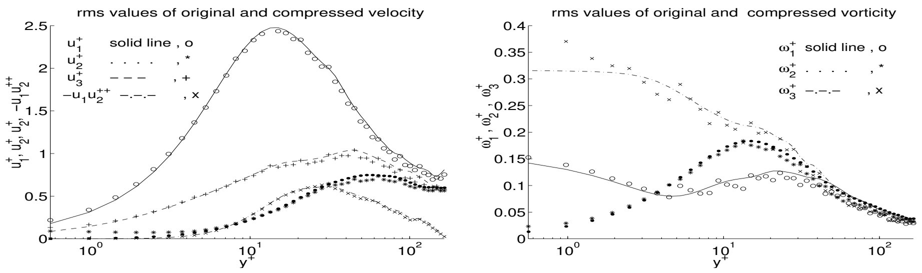

Figure 8: Exp. II/ANISO/PRW/H1 ; CR=63

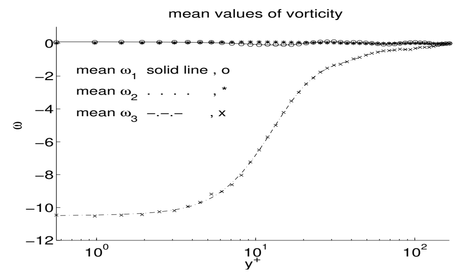

Figure 9: Exp II/ANISO/PRW/H1 ; CR=63

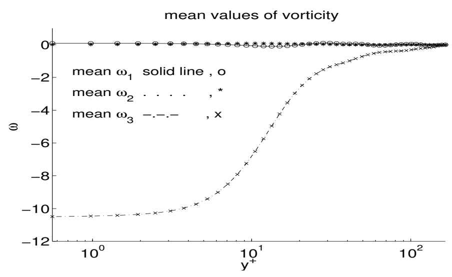

Figure 10: Exp II/ANISO/PRW/SPG ; CR=47

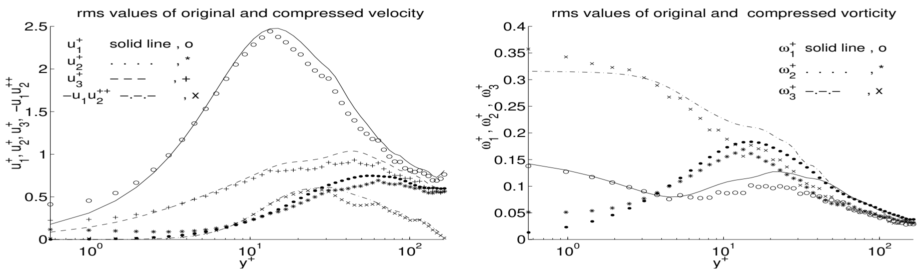

Figure 11: Exp. II/ANISO/PRW/SPG ; CR=47

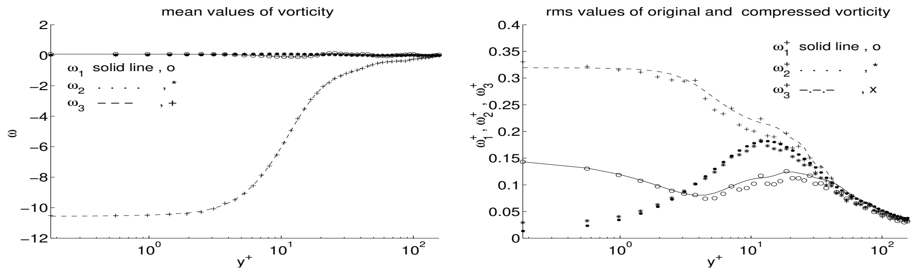

Figure 12: Exp. III /ANISO/PRW/L2 ; CR=63

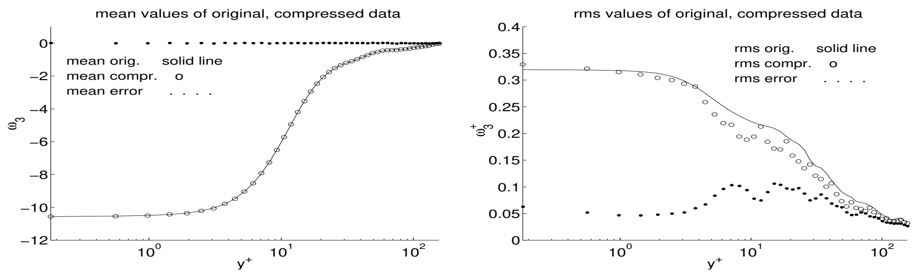

Figure 13: Exp. I /ANISO/PRW/L2 ; CR=63

coarser level $j-1$.

5.1 Hash Table Data Structures

The scattered distribution of active coefficients and their small number compared to all coefficients forbid the use of ordinary arrays for storage. There have to be found other methods. The problem of efficient storage of scattered data is known in computer sciences for decades and there are some well established techniques, which provide more or less efficient storage and fast access/insert/delete of data. The atoms access/insert/delete are exactly what we require in Table 3. Some of these common techniques are lists, tree data structures or hash table data structures [22]. Until recently, state-of-the-art numerical codes used tree data structures only. Instead of that, in [18], [19] the use of hash tables was proposed. Let N be the number of indices in Λ_n, which in applications will be of order $10^4 ... 10^6$, then tree data structures have a $\mathcal{O}(\log(N))$ worst/average case complexity of the atoms access/insert/delete. Especially for often used operations, as the addition of two functions by means of adding their coefficients, the overhead for accessing the data is considerable or even dominating compared to the actual numerical operation.
Hash tables solve this problem. In comparison to the tree approach hash table addressing gives more or less direct access to the data stored, i.e. it is proven to possess a $\mathcal{O}(1)$ complexity with a low constant if a statistical data distribution

given the approximation $\mathbf{u}_{\Lambda_0},\ p_{\Lambda_0}$ of $(\mathbf{u},p)(t_0=0)$	
	compute $\mathbf{u}^{n+1}_{\Lambda_n},\ p^{n+1}_{\Lambda_n}$ from $\mathbf{u}^{n}_{\Lambda_n},\ p^{n}_{\Lambda_n}$
	mark indices (j,l) of important coefficients $(\mathbf{u}^{n+1},p^{n+1})^j_l$ and their neighbours in space and scale for retaining or insertion and mark small coefficients for deletion to obtain new index set Λ_{n+1}
	insert/delete coefficients to obtain $\mathbf{u}^{n+1}_{\Lambda_{n+1}},\ p^{n+1}_{\Lambda_{n+1}}$
$n=n+1$	

Table 3: Basic spatial adaptive algorithm

is assumed. Furthermore lower additional storage overhead for logical connectivities are required. Let us briefly describe this appealing idea.
Each index (j,l) is mapped by a certain hash function h to a hash key, which is used as an address in the hash table. The hash table is an m-element array of pointers to m different lists, in which all the required data (j,l,X^j_l) is stored. This basic approach is depicted in Figure 14. The lists are necessary, since the

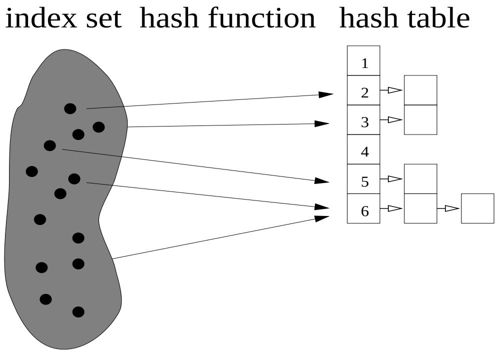

Figure 14: Hashing

number of admissible indices (j,l) is much larger than the number N of active coefficients. Thus h can not be injective. Usually m is of order N to provide short lists and fast average access to the data. As an example we have stored the coefficients with respect to the hierarchical nodal basis of the function shown in Figure 15 with a hash table. Figure 16 depicts the distribution of the lengths of the lists. Apparently the hash function used in this example leads to a very broad distribution of the hash keys and the lists are almost of optimal length.
It seems that hash tables first have been used in adaptive PDE solvers and in the context of adaptive multigrid solvers by [18].

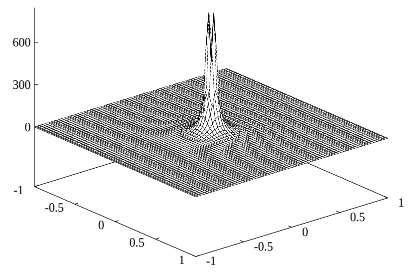

Figure 15: Example

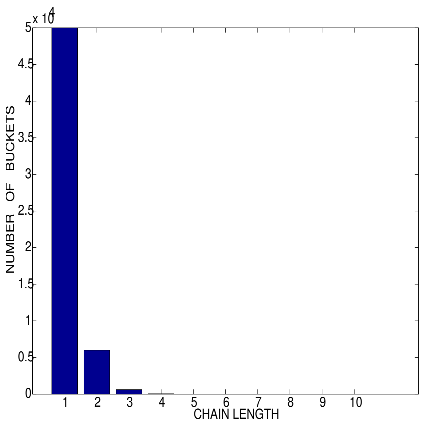

Figure 16: list lengths

5.2 Parallelization with Space-filling Curves

Besides the reduction of the complexity of the physical problem, e.g. by multiscale methods, nowadays one has the opportunity to distribute large problems onto several processors. This speeds up the computations. Another reason might be the huge amount of memory required for the solution, which is not present on a single processor machine. To tackle a new class of problems (think of a great challenge project) it is necessary to combine both approaches.
Since the scatteredness of the data in our case rules out the use of 3D array data structures, the usual domain decomposition technique is no longer advisable. A moving region of strong variations in the flow, for a fixed domain decomposition, would cause a very high load for some processors, while others are waiting and wasting precious computing time. Thus we have to find other load balancing techniques, which hopefully fit to the hash table storage technique. A very promising approach has been introduced in [18], [19]. We shall briefly describe this idea for the bivariate case.

In a first step the indices $(j,l) = ((j_1,j_2),(l_1,l_2))$ are mapped to unique nodes $(x,y) \in [0,1]^2$ of a certain finest grid. Usually this will be something like $(l_1 2^{-j_1}, l_2 2^{-j_2})$. Hence, the node is located within the support of ψ_l^j, which nicely corresponds to the localization of the basis functions. Then, in a second step the values (x,y) are mapped to a value $s \in [0,1]$. This is done by a space-filling curve, which provides us with an injective, continuous mapping $f : [0,1]^2 \bigcap Q \times Q \mapsto [0,1]$. Here, Q is the set of numbers with finite binary expansion.

$$(j,l) \quad \overset{g \text{ (injective)}}{\longleftrightarrow} \quad (x,y) \in [0,1]^2 \quad \overset{f \text{ (injective)}}{\longleftrightarrow} \quad s \in [0,1]$$

The evaluation of both mappings g and f requires only some cheap operations and is fast. After applying these two steps for each $(j,l) \in \Lambda_n$, we have a set

$\{s(j,l) \mid (j,l) \in \Lambda_n\}$. These values are increasingly ordered.

$$s_1 < \ldots < s_N \quad N = \#\Lambda_n$$

For the load balancing the indices (j,l), and the data associated to them, are distributed among the p processors in the following way.

$$\begin{array}{lllll} \text{P1:} & \{(j,l) \mid s_1 & \leq s(j,l) \leq & s_{N/p-1}\} \\ \vdots & \vdots & & \vdots \\ \text{P}p\text{:} & \{(j,l) \mid s_{(p-1)N/p} & \leq s(j,l) \leq & s_N\} \end{array}$$

Figure 17 shows an example of a space-filling curve and the distribution of the nodes onto three processors in the non adaptive case.

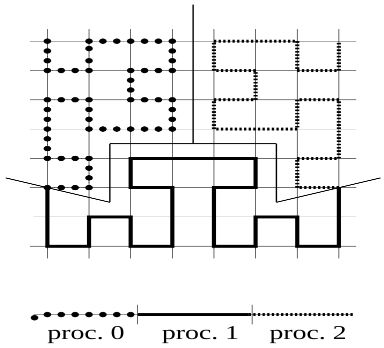

Figure 17: Load balancing using space-filling curves

Clearly this technique leads to an optimal load balancing, while the volume of communication depends on the boundaries of the partitions. These boundaries may sometimes be kinky, depending on the data, and are certainly not optimal, but are of reasonable size. In total, the load balancing is very cheap, parallelizes well and thus can be applied in each cycle of algorithm 3.
In addition there are great similarities between this load balancing technique and hash table storage. Both techniques lead to, in some sense, linear ordered data. E.g. simple modifications of f are good candidates for the local hash function on each processor.

6 Conclusion

The goal of this paper has been two-fold: first to test and extend the wavelet techniques that have been developed for isotropic two-dimensional turbulence, and secondly to evaluate the possibility of using a similar approach for three-dimensional turbulence with boundaries. The results of the first investigation suggest a new way of modelling high Reynolds number two-dimensional turbulence, while the second investigation has helped to clarify the general properties of a multiscale adaptive wavelet-based method for solving the three-dimensional Navier-Stokes equations (3D-NSE) at high Reynolds number.

The adaptive wavelet technique for the turbulent 2D-NSE was found to be highly accurate and to use far fewer active modes than standard or nonlinearly filtered pseudo-spectral techniques. Furthermore, the number of active modes remains constant, even during periods of intensely nonlinear interaction with strong gradients. This suggests that the wavelet representation is well-suited to the dynamics of the 2D-NSE at high Reynolds number.

A new, more physical, forcing technique based on the wavelet representation was proposed and tested. In this method the strongest wavelet coefficients (corresponding to the coherent vortices) are reinforced at each time step. The forcing models the way turbulence is generated in real flows by vorticity production via an instability at the boundaries. This method produces stationary statistics, and can be used to provide a more realistic forcing in simulations of turbulent flows. The method should also work in three dimensions.

Wavelet analysis of two-dimensional turbulence showed that the flow may be divided into a coherent part (the vortices) with non-Gaussian one-point vorticity statistics and an incoherent part (the background flow) with Gaussian one-point statistics. The threshold, dividing the weak and strong wavelet modes, is based on the variance of the vorticity field and the number of points and is thus objective (and constant if the simulation is stationary). The coherent part of the flow represents only 1% of the total number of modes for a resolution of 128^2 and will decrease for higher resolution.

The fact that the background has Gaussian statistics suggests that it could be easily modelled, and in this case the adaptive wavelet method would only used for the coherent vortices. A study of the vorticity filaments which make up the background showed that they should be stabilized by the coherent vortices they are associated with, provided the density of vortices is not too low. However, the filaments do have an effect on the flow since they shield the vortex they surround from the deformation induced by the strain of neighbouring vortices. It remains to be seen whether this effect significantly alters the turbulence dynamics.

In summary, these investigations have established the usefulness and accuracy of wavelet techniques for analyzing and solving the 2D-NSE at high Reynolds numbers. The results also suggest a new way of modelling two-dimensional turbulence. The dynamics of the coherent vortices could be found by directly solving the 2D-NSE using an adaptive wavelet technique, while the effect of the background could be modelled using a simple statistical model (since the background vorticity field has Gaussian one-point statistics).

The second goal of the paper was to evaluate the potential of various multiscale methods for the solution of the three-dimensional 3D-NSE. We showed that, even with some simple error estimators and simple prewavelets, reasonable approximations of the original turbulence data are possible for relatively high compression rates of about 60 at resolution $192 \times 96 \times 192$.

Furthermore, it turned out that for the solution of the 3D-NSE, at least for flows which are governed by boundaries, the formulation in primitives variables seems to be more efficient compared to the formulation in terms of vorticity. In the fifth section we presented efficient methods for the implementation of an fully adaptive solver. These methods produce very efficient data storage and a cheap, but reasonably effective, load balancing technique.

The results presented here have demonstrated the usefulness of adaptive wavelet techniques for the 2D-NSE at high Reynolds numbers, and have laid the foundations of an equivalent technique for the 3D-NSE. The next stages in two and three dimensions are, respectively, the development of a highly efficient turbulence model (where only the coherent vortices are calculated exactly) and the implementation of a solver for the 3D-NSE.

7 Bibliography

[1] K. Akselvoll, P. Moin *An efficient method for temporal integration of the Navier-Stokes equations in confined axisymmetric geometries*; J. Comp. Phys. 125 (1996), pp. 454

[2] G.K. Batchelor *The theory of homogeneous turbulence*; Cambridge University Press (1953)

[3] D. Bonn, Y. Couder, P. Van Dam *From small scales to larges scales in 3D turbulence*; Phys. Rev E 47 (1993)

[4] H.J. Bungartz *Sparse Grids and their Application for the adaptive Solution of the Poisson Eqn. (in german)*; Thesis TU München (1992)

[5] P. Charton, V. Perrier *A pseudo-wavelet scheme for the two-dimensional Navier-Stokes equation*; Comp. Appl. Math. 15/2 (1996), pp.139–160

[6] A. Cohen, I. Daubechies, P. Vial *Wavelets on the Interval and Fast Wavelet Transforms*; Appl. Comp. Harm. Anal. 1 (1993)

[7] W. Dahmen *Wavelet and Multiscale Methods for Operator Equations* ; Acta Numerica (1997)

[8] I. Daubechies *Orthonormal bases of compactly supported wavelets* ; Comm. Pure Appl. Math. (1988)

[9] I. Daubechies *Orthonormal bases of compactly supported wavelets II* ; SIAM J. Math. Anal. 24 (1993)

[10] D. L. Donoho *Wavelet Shrinkage and W.V.D. – A Ten-Minute Tour*; Technical Report 416 (1993)

[11] M. Farge et.al. *Wavelets and Turbulence*; Proceedings of the IEEE Vo. 84 No. 4 (1996)

[12] M. Farge, K. Schneider, N. K.-R. Kevlahan *Coherent structure eduction in wavelet-forced two-dimensional turbulent flows*; To appear in *Dynamics of slender vortices* (ed. E. Krause). Cambridge University Press.

[13] J. Fröhlich, K. Schneider *An adaptive wavelet-vaguelette algorithm for the solution of PDEs.*; J. Comput. Phys. 130 (1997) pp.174–190.

[14] J. Fröhlich, K. Schneider *Numerical simulation of decaying turbulence in an adapted wavelet basis*; Appl. Comput. Harm. Anal. 3 pp. 393-397

[15] M. Griebel, F. Koster *Remarks on orthogonal wavelets on the interval*; Preprint Univ. Bonn , SFB 256 (1998)

[16] M. Griebel, P. Oswald *Tensor product type subspace splittings and multilevel iterative methods for anisotropic problems*; Advances in Comp. Math. (1995)

[17] M. Griebel, W. Huber *Turbulence Simulation on Sparse Grids Using the Combination Method*; TU München SFB Bericht Nr. 342/19/94 (1994)

[18] M. Griebel, G. Zumbusch *Hash-storage techniques for adaptive multilevel solvers and their domain decomposition parallelization*; Proc. of the 10th Int. Conf. on Domain Decomposition Methods, Boulder, USA (1997)

[19] M. Griebel, G. Zumbusch *Parallel multigrid in an adaptive PDE solver based on hashing* ; Parallel Computing (ParCo 97), Bonn, (1997)

[20] H.J. Kaltenbach *Unpublished data*; TU Berlin (1997)

[21] N. K.-R. Kevlahan,M. Farge *Vorticity filaments in two-dimensional turbulence: creation, stability and effect*; J. Fluid Mech. 346 (1997) pp.49–76.

[22] D.E. Knuth *The Art of Computer Programming 3*; Addison Wesley (1973)

[23] R.H. Kraichnan *A theory of turbulence dynamics*; II. Symposium on Naval Hydrodynamics (1958), Office of Naval Research, Washington DC, ACR-38

[24] A. Leonard *Computation of 3D incompressible flows with vortex elements*; Ann. Rev. Fluid Mech. 17 (1985), pp.523

[25] J. McWilliams *The emergence of isolated coherent vortices in turbulent flow*; J. Fluid Mech. 146 (1984), pp.2–43.

[26] Y. Meyer *Ondelettes sur l'intervalle*; Rev. Mat. Iberoamericana 7 (1992)

[27] P. Monasse, V. Perrier *Orthonormal Wavelet bases adapted for PDE with boundary conditions*; Preprint LMD Paris (1995)

[28] P. Oswald *Multilevel finite element approximation*; Teubner Verlag (1994)

[29] K. Schneider, M. Farge *Wavelet forcing for numerical simulation of two-dimensional turbulence*; C. R. Acad. Sci. Paris 325 IIb (1997), pp. 263–270.

[30] K. Schneider, N. K.-R. Kevlahan, M. Farge *Comparison of an adaptive wavelet method and nonlinearly filtered pseudo-spectral methods for two-dimensional turbulence*; Theoret. Comput. Fluid Dynamics **9** (1997) pp.191–206.

[31] E.D. Siggia *Numerical study of small-scale intermittency in 3D turbulence*; J. Fluid Mech. 107 (1981), pp. 375-406

[32] W. Sweldens *The Lifting Scheme: a construction of second generation wavelets*; SIAM J. Math. Anal. (1997)

[33] W. Sweldens, R. Piessens *Quadrature Formulae and Asymptotic Error Expansions*; SIAM J. Num. Anal. 31 (1994)

[34] A. Vincent, M. Meneguzzi *The dynamics of vorticity tubes in homogeneous turbulence*; J. Fluid Mech. 258 (1994), pp.245-254

[35] C. Zenger *Sparse Grids*; Notes on Numerical Fluid Mechanics Vol. 31, Vieweg (1991)