A Parallel Spectral Fourier - Nonlinear Galerkin Algorithm for Simulation of Turbulence

appeared in Numerical Methods for Partial Differential Equations, pp. 699-715, 1997.

A. Averbuch[1], L. Ioffe[2], M. Israeli[2], and L. Vozovoi[1]

[1] School of Mathematical Sciences, Tel Aviv University, Tel Aviv 69978, Israel

[2] Faculty of Computer Science, Technion, Haifa 32000, Israel

Abstract

We present a high-order parallel algorithm which requires only the minimum interprocessor communication dictated by the physical nature of the problem at hand. The parallelization is achieved by domain decomposition. The discretization in space is performed using the Local Fourier Basis method. The continuity conditions on the interfaces are enforced by adding homogeneous solutions. Such solutions often have fast decay properties which can be utilized to minimize interprocessor communication. In effect, the predominant part of the computation is performed independently in the subdomains (processors) or using only local communication.

A novel element of the present parallel algorithm is the incorporation of a Nonlinear Galerkin strategy to accelerate the computation and stabilize the time integration process. The basic idea of this approach consists of decomposition of the variables into large scale and small scale components with different treatment of these large and small scales.

The combination of the Multidomain Fourier techniques with the Nonlinear Galerkin (NLG) algorithm is applied here to solve incompressible Navier-Stokes equations. Results are presented on direct numerical simulation of two-dimensional homogeneous turbulence using the NLG method.

1 Introduction

Direct numerical simulation of turbulent flows at high Reynolds number $Re \sim 10^6 - 10^9$ requires considerable computational resources due to the extreme space and time resolution (the computational requirements scales like $Re^{9/4}$). High order methods, in particular spectral methods, are preferred for turbulence computation as they converge rapidly with the number of degrees of freedom. As a consequence, for moderate to high accuracy a smaller number of degrees of freedom may be needed, especially in multi-dimensional problems. It has been also proven that high order methods are beneficial for the long term integration necessary to attain an asymptotic turbulent regime.

Large scale scientific computing can be realized on massively parallel computers. However, the global nature of spectral methods makes these methods difficult to parallelize. Current parallel spectral algorithms require massive data transfers and global communication with the concomitant communication and synchronization bottlenecks. This can degrade seriously the speedup achieved by the parallelization.

In this paper we present a high-order parallel algorithm which requires only the minimum inter-processor communication dictated by the physical nature of the problem at hand. The parallelization is achieved by domain decomposition while the spatial discretization in subdomains is performed with the Local Fourier Basis method. Thus the global data exchange of conventional parallel Fourier methods is avoided while saving the spectral convergence rate by decomposition of forcing functions into smooth local pieces.

An important feature of our algorithm is that it can incorporate the Nonlinear Galerkin (NLG) method to speed up the computation and reduce the amount of data to be transfered among the processors (domains).. This approach has a sound mathematical basis in the theory of inertial manifolds and dynamical systems [6, 11]. Inertial manifolds have been introduced as a means for describing the interaction of the low and high frequency components of a flow. Indeed when it exists an inertial manifold describes the slaving of the high frequencies by the low frequencies. Hence, it is one of the motivations of inertial manifolds to produce a tool for studying the long time behavior of nonlinear dissipative systems and uncovering some structure in the real or apparent chaos of nonlinear dynamics.

More generally, inertial manifolds (IM) and approximate inertial manifolds yield exact or approximate slaving laws of the small scale component of a flow by its large scale component. Exact and approximate IM produce a differentiated treatment of different components of a flow and they are in essence multilevel methods. With that respect IM are related to the developing chapter of scientific computing constituted by the multilevel methods.

The NLG method is a multilevel method stemming from the theory of inertial manifolds. It is based on a decomposition of the unknown function into its small scale and large scale components. It is interesting to observe that the small scale or high frequency components of a flow accounts for a large portion of the total number of unknowns while it carries only a small fraction of the total energy or of the magnitude of the unknown function. This can lead to serious computational difficulties. The multilevel methods try to address this computing problem in relation with the information on the dynamics deduced from the study of the attractor and its approximation.

The purpose of the present paper is to demonstrate that a certain version of the NLG approach can be easily incorporated into parallel computation. Particularly, the multidomain local Fourier (MDLF) method can be combined with the NLG approach in a natural way as it performs the required decompositions into long and short waves.

The paper is organized as follows. In section 2, we describe the Multidomain Local Fourier algorithm. It is applied for parallel computation of the incompressible Navier-Stokes equations. In section 3, we apply this algorithm on direct numerical simulation of the two-dimensional decaying turbulence. In section 4, we introduce a new version of the MDLF algorithm which incorporates the Nonlinear Galerkin method for the different treatment of small and large eddies. Numerical results that were derived by the application of the novel MDLF-NLG algorithm are discussed in section 5.

2 Multidomain Local Fourier Algorithm

Formulation of the problem -

An important application of the present Multidomain Local Fourier (MDLF) method is the solution of the incompressible Navier-Stokes equations.

Governing Equations:

$$\frac{\partial \mathbf{v}}{\partial t} = Re^{-1}\nabla^2\mathbf{v} + \mathbf{N}(\mathbf{v}) - \nabla\Pi + \mathbf{f} \text{ in } \Omega. \tag{2.1}$$

Here $\Omega \subset R^d$, $d = 2, 3$, $\mathbf{v}(\mathbf{x}, t) = (u, v, w)$ is the velocity, subject to the incompressibility constraint

$$\nabla \cdot \mathbf{v} = 0 \qquad \text{in } \Omega, \tag{2.2}$$

Π is the total pressure, $\mathbf{f}$ is the forcing term, and Re is the Reynolds number. The nonlinear term is written in the rotational form

$$\mathbf{N}(\mathbf{v}) = \mathbf{v} \times (\nabla \times \mathbf{v}). \tag{2.3}$$

Computational Domain and Boundary Conditions.
The computational domain is the rectangle $\Omega = \{0 \leq x < 2\pi, 0 \leq y < 2\pi\}$. The periodic boundary conditions in both directions are considered:

$$\mathbf{v}(x + 2\pi, y) = \mathbf{v}(x, y + 2\pi) = \mathbf{v}(x, y). \tag{2.4}$$

The numerical solution of the problem (2.1)-(2.4) with specified boundary conditions requires discretization in both time and space.

Discretization in Time -

The discretization in time is performed via the 3d-order splitting algorithm of [10]:

$$\frac{\hat{\mathbf{v}} - \sum_{q=0}^{2} \alpha_q \mathbf{v}^{n-q}}{\Delta t} = \sum_{q=0}^{2} \beta_q \mathbf{N}(\mathbf{v}^{n-q}), \tag{2.5}$$

$$\frac{\hat{\hat{\mathbf{v}}} - \hat{\mathbf{v}}}{\Delta t} = -\nabla\Pi^{n+1}, \quad \nabla^2\Pi^{n+1} = \frac{1}{\Delta t}\nabla \cdot \hat{\mathbf{v}}, \tag{2.6}$$

$$\frac{\gamma_0 \mathbf{v}^{n+1} - \hat{\hat{\mathbf{v}}}}{\Delta t} = Re^{-1}\nabla^2\mathbf{v}^{n+1}. \tag{2.7}$$

where the coefficients γ_0 and α_q are the standard coefficients of the implicit stiffly stable schemes corresponding to the order J_i. The coefficients β_q for the explicit contributions are computed by the method of undetermined coefficients and employing Taylor series expansions. We used a 3-d order scheme $J_i = J_e = 3$ with the coefficients: $\gamma_0 = 11/6, \alpha_0 = 3, \alpha_1 = -3/2, \alpha_2 = 1/3, \beta_0 = 3, \beta_1 = -3, \beta_2 = 1$. It consists of an explicit advection step (2.5), a global pressure adjustment for incompressibility (2.6) and an implicit viscous step (2.7).

Semi-implicit schemes of this type are associated with much less severe restriction on the time step, $O(N^{-1})$, than fully explicit schemes, $O(N^{-2})$, where $2\pi/N$ is the mesh size.

As a result of the splitting procedure, the time-dependent problem is reduced to the solution (for each time step) of a sequence of two types canonical elliptic equations: the modified Helmholtz equation

$$\nabla^2 u - \lambda^2 u = f(x,y), \tag{2.8}$$

and the Poisson equation

$$\nabla^2 u = g(x,y), \tag{2.9}$$

The parameter λ in (2.8) is related to the time-stepping increment, $\lambda^2 \propto Re/\Delta t$.

Domain Decomposition: Discretization in Space -

First we briefly describe the 1-D algorithm as applied to Eq. (2.8), then we show the extension to the 2-D case. The detailed description can be found in [7, 8, 12].

Algorithm in 1-D

1. Divide the computational interval $[0, L]$ into P subdomains, $l = L/P$, and discretize the local problems on uniform grids of collocation points (with different resolutions in each subdomain, if desired).
2. Decompose the source function $f(x)$ into smooth local pieces $f^{(n)}(x)$ using a collection of overlapping window functions $b(x)$:

$$f(x) = \sum_{k\in Z} c_k \phi_k, \qquad \phi_k = \sum_{n=1}^{P} \frac{2}{I_n} b_n(x) \sin\frac{k\pi(x-\bar{x}_n)}{I_n} \tag{2.10}$$

   where I_n is the size of a $n-$th subdomain, $\bar{x}_n$ are the locations of the interior boundaries (interfaces), and c_k are constant amplitudes. Each of the window functions $b_n(x)$ is equal to unity inside its respective subdomain and decays smoothly outwards over a distance of 2ϵ from both sides. The functions $\{\phi_k(x)\}$ present a particular form of the *Local Trigonometric Basis* introduced in [3].
3. Integrate the local problems on extended intervals $l + 2\epsilon$ with smooth source functions in the right hand side. The values of the computed particular solutions $u_p^{(n)}$ will have jumps through the interfaces.
4. Correct the solutions in each subdomain in order to enforce the continuity of $u(x)$ and $u'(x)$ at the interfaces then the higher derivatives will be matched automatically due to Eq. (2.8). It can be done by adding the properly weighted homogeneous solutions

$$\begin{aligned} u &= \bigcup_{n=1}^{P} u^{(n)}, \\ u^{(n)} &= u_p^{(n)} + A_n h_+^{(n)} + B_n h_-^{(n)}. \end{aligned} \tag{2.11}$$

These solutions are two exponential functions $h_-^{(n)} = e^{-\lambda x}$, $h_+^{(n)} = e^{-\lambda(l_n - x)}$, $0 < x < l_n$, which decay into a subdomain. Since $\lambda \gg 1$ for a small enough time step, the influence of remote interfaces on the coefficients A_n, B_n for each particular n is negligible. Therefore, these coefficients can be found in terms of jumps at one corresponding interface. This requires only the local communication between neighboring subdomains[1].

The error caused by the use of an approximate local matching instead of the exact global matching is $\varepsilon = O(e^{-\lambda l})$, and it is *independent* of the resolution N. For example, for $P = 4$, $l = 2\pi/P \approx 1.6$ and $\Delta t = 0.04$ we have $\lambda \approx 10$. For example, for $Re = 1$ the approximation error due to the use of the local matching is 10^{-7}, and it is even smaller for larger Re. Note that the above restriction on the size Δt due to the local matching might be less severe than that imposed by the use of an explicit scheme in time. In the last case the stability constraint depends on the grid size like $\Delta t Re^{-1} \leq h^2/2$ where $h = 2\pi/N$. For a realistic resolution, e.g. $N = 128$, and the Reynolds number $Re = 1$ we have $h = 2\pi/N \approx 0.05$, $\Delta t \leq 0.0013$. In such a case the use of a semi-implicit scheme, combined with the local matching procedure, is still advantageous over explicit schemes.

Algorithm in 2-D (3-D): Decomposition into Strips (Slabs)

Here we describe the 2-D approach and the 3-D version follows as a natural extension (see [1] for details).

1. Decompose the computational rectangular region into parallel strips: $\Omega = \cup_{n=1}^{P} \omega_n$.

2. Apply the FFT in the periodic direction y (along the strips) to obtain a collection of *uncoupled* 1-D ODEs for the Fourier coefficients $\hat{u}_k(x)$:

$$\frac{d^2 \hat{u}_k^{(n)}}{dx^2} - \lambda_k^2 \hat{u}_k^{(n)} = \hat{f}_k^{(n)}(x) \qquad \text{in } \omega_n, \tag{2.12}$$

where $k = -\frac{N_y}{2}, \ldots, \frac{N_y}{2}$, $\lambda_k^2 = \lambda^2 + k^2$ for the modified Helmholtz equation and $\lambda_k^2 = k^2$ for Poisson equation.

3. Solve the problems (2.12) by the previous 1-D algorithm. The modified Helmholtz equation is solved using the local matching procedure on the interfaces. In the case of Poisson equation, $\lambda = 0$, $\lambda_k^2 = k^2$, the global matching is required only for the long waves, $k \leq k_*$, whereas the short waves, $k \geq k_*$, can be treated by using local matching on the interfaces. The cut-off wave number k_* should be chosen in accordance with the prescribed accuracy.

In order to estimate the potential of this approach we present the following numerical tests. We solve the 2-D Poisson equation with the source function $f(x, y)$ corresponding to the solution $u_{ex} = \cos ky \sum_{m=1}^{M} \cos(mx + \alpha_m)/m$ where the phases α_m are chosen randomly. Table 1 displays the maximum relative error $\varepsilon_k = |u - u_{ex}|_{max}/u_{max}$ for several numbers of globally matched harmonics, k (the error of the "full" global matching in this case is approximately $\varepsilon \sim 10^{-12}$).

k	3	5	7	9
ε_k	5.4 (-3)	5.7 (-4)	6.0 (-5)	6.3 (-6)

Table 1: The maximum numerical error when using the global matching for the first k harmonics; $P = 8, N_x = 64 \times P, N_y = 64$

We can see that using the global matching only for the first 9 long waves, while treating locally the rest 21 shorter waves, we guarantee the accuracy $\varepsilon_k \leq 10^{-5}$. The relative amount of globally matched waves, $2k^*/N_y$, becomes smaller as the resolution in the periodic direction increases. Note that the accuracy of the local matching relies on the localization properties of the boundary Green's functions of the Helmholtz operator. Therefore, the results in Table 1 depend slightly only on the particular form of the source function $f(x, y)$.

Decomposition into Cells.

The algorithm in this case is more complicated than the previous one. For details see [12].

Parallel Test -

The parallel test is performed on a collection of Digital Alpha workstations (Alpha-farm) with 128 MB per processor. They are connected via Giga-switch communication network. This machine is programmed by using the PVM which is kind of "universal" parallel software package. The computational domain is a 3-D periodic box, decomposed into "slabs" ($P = 2 - 10$). For the resolution 128^3 the effect of using the local-global matching, when solving Poisson equation, results in 2-3 times reduction of the executable time as compared to the "fully" global matching.

When solving the complete Navier-Stokes Eqs. (2.1)-(2.3)([1]) the implicit viscous step is always computed using the local matching. The use of the global-local matching in the pressure step gives about $15 - 20\%$ saving in the CPU time as compared to the global matching. This effect is expected to be even more pronounced for a larger number of processors. Detailed discussion of the parallel algorithm and its performance is given in [1].

- **Test of Accuracy** .

The following example testifies the accuracy of the MDLF algorithm. We consider the exact solution of Eqs. (2.1)-(2.4)

$$v_x = \frac{\partial \psi}{\partial y}, v_y = -\frac{\partial \psi}{\partial x},$$

$$\psi(x, y) = g(t)e^{(\cos \omega x)(\cos \omega y)}$$

$$g(t) = \frac{1}{40}\left(\frac{1}{10}e^{(sin 2t - 3 sin 2\pi t)} + cos(2\sqrt{2}t)\right) + c$$

with $Re = 10^2, \omega = 5,\ c = 0.075$ and the forcing function $f(x, y, t)$ computed accordingly (this solution was proposed in [9] to demonstrate the capability of the multilevel Nonlinear Galerkin method for the long-time numerical integration). The evolution of the maximum relative error in time is shown in Fig. 2; $\Delta t = 5 \cdot 10^{-3}$ and 10^{-4}, the strip decomposition with $P = 4$ is applied, the space resolution is 64×64 . The error remains small on a long time scale and it decreases as the size of the time step Δt becomes smaller (the spatial error for this example is $O(10^{-11})$ can be found with $g(t) =$const(t)).

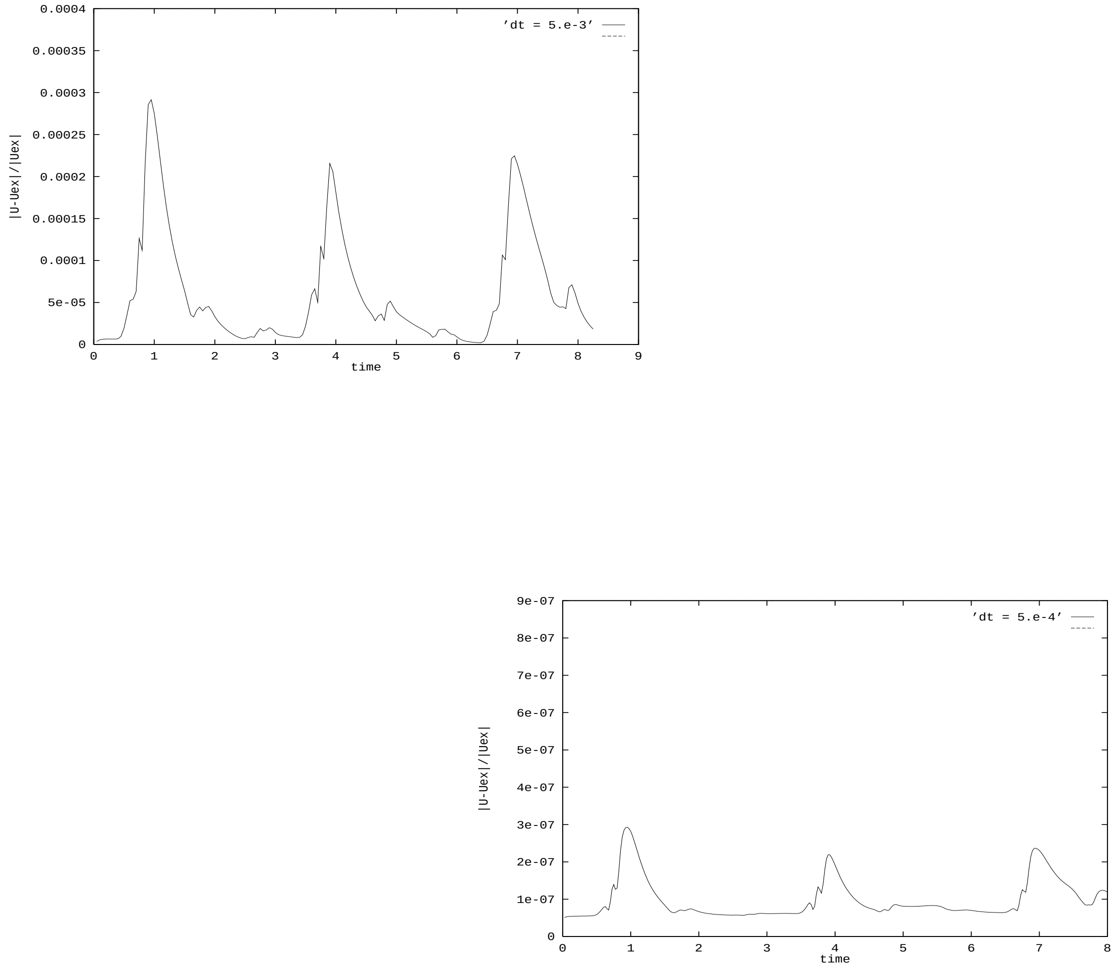

Figure 1: Evolution of the relative maximum error in time for $\Delta t = 5 \cdot 10^{-3}$ (left plot) and $\Delta t = 5 \cdot 10^{-4}$ (right plot).

3 Parallel Simulation of 2-D Decaying Turbulence

In this section we present results on numerical simulation of the homogeneous (space periodic) decaying turbulence using the above algorithm. The external volume force **f** in Eq. (2.1) is equal to zero. The initial velocity field in the Fourier domain is chosen as in [2]:

$$\hat{\psi}(k) = |\hat{\psi}(k)| e^{i\theta(k)}$$

$$|\hat{\psi}(k)| = \begin{cases} \frac{C}{k\sqrt{1+(k/6)^4}}, & |k| < 20.5 \\ 0, & |k| \geq 20.5 \end{cases} \tag{3.1}$$

where $\psi(x,y)$ is the stream function defined by

$$\mathbf{v} = \{u, v, 0\}, \quad u = \frac{\partial\psi}{\partial y}, \quad v = -\frac{\partial\psi}{\partial x}. \tag{3.2}$$

The phase $\theta(k)$ is chosen randomly in the interval $[0, 2\pi]$.

Now we introduce the vorticity

$$\mathbf{Z} = (0, 0, \zeta), \qquad \zeta = -\Delta\psi,$$

and the two norms:

$$|\mathbf{v}|_2 = \left(\int_\Omega |\mathbf{v}(\mathbf{x}, t)|^2 d\mathbf{x}\right)^{1/2}$$

and

$$||\mathbf{v}|| = \left(\int_\Omega |\nabla\mathbf{v}(\mathbf{x}, t)|^2 d\mathbf{x}\right)^{1/2}$$

Notice that these norms are related to fundamental physical quantities: the kinetic energy of the velocity field

$$e(\mathbf{v}) = \frac{1}{2}|\mathbf{v}|^2,$$

and, in the case of periodic boundary conditions, the enstrophy

$$ens(\mathbf{v}) = \frac{1}{2}||\mathbf{v}||^2 = \frac{1}{2}|\mathbf{Z}|_2^2.$$

The computed solution (the contour plot of vorticity ζ) is shown in Fig. 3 at $t = 20$ and $t = 100$; $\Delta t = 0.01$, $Re = 10^4$, the resolution 128×128, the number of processors $P = 4$.

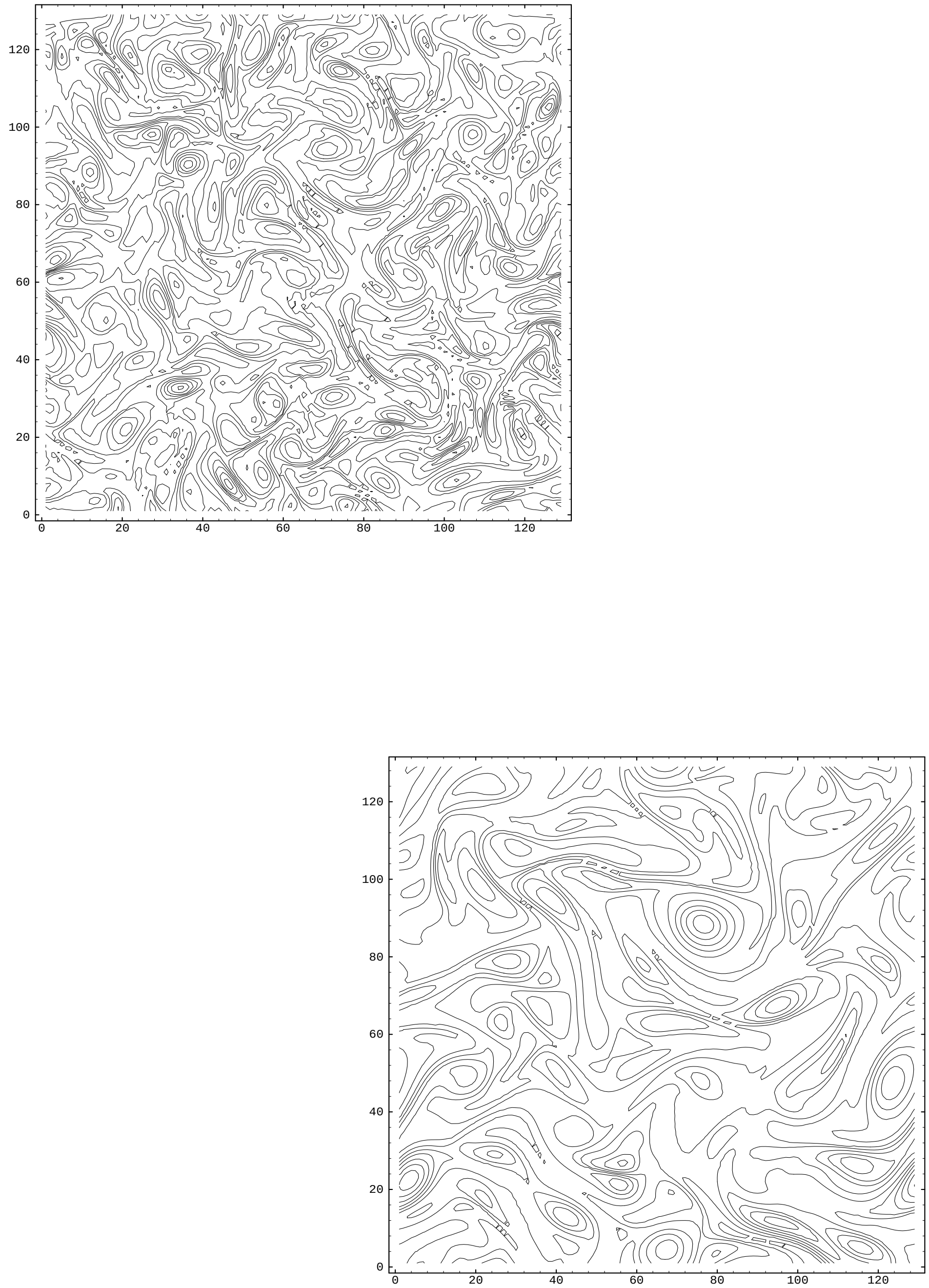

Figure 2: Contour plots of vorticity at $t = 20$ (left plot) and $t = 100$ (right plot).

Practically the same plots of the vorticity are obtained using the resolution 256 × 256. The results depend only slightly on the size of the time step in the range $\Delta t = 10^{-4} - 10^{-1}$. This is surprising, as one might think that the temporal truncation error would dominate. This effect is discussed in [2].

The enstrophy spectra are plotted in Fig. 3 for $t = 20$ and 100. Naturally, the small scales dissipate faster than the large scales.

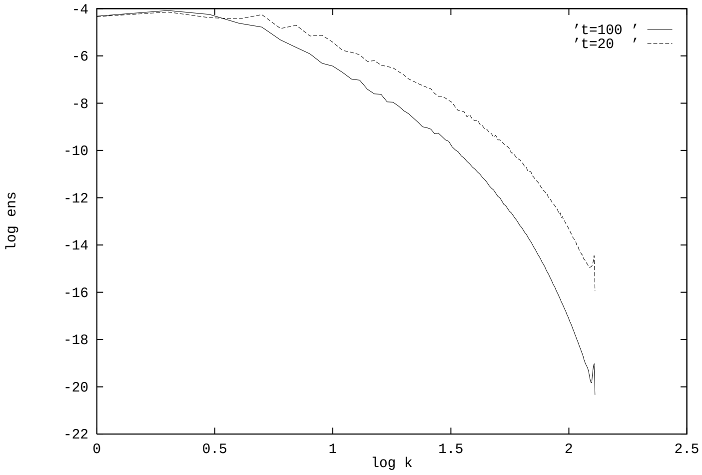

Figure 3: The enstrophy spectra at $t = 20$ and $t = 100$; resolution 256×256, number of subdomains $P = 4$.

The plots in Fig. 3 are very similar (regarding the flow structure, the sizes of the eddies) to those obtained in [2] with a pseudo-spectral Fourier method in a single domain (they are not expected to be identical since the phase $\theta(k)$ is a random field). These results indicate that the MDLF algorithm allows a stable time integration at high Reynolds numbers with adequate resolution of small features in the flow.

For a given Reynolds number, the resolution must be large enough to provide dissipation of energy at small scales. In Fig. 3 the vorticity is plotted for the resolution 64×64 and 128×64 (128 points across and 64 points along the strips). The large amplitude in high wave numbers in small areas is typical of cases with insufficient resolution.

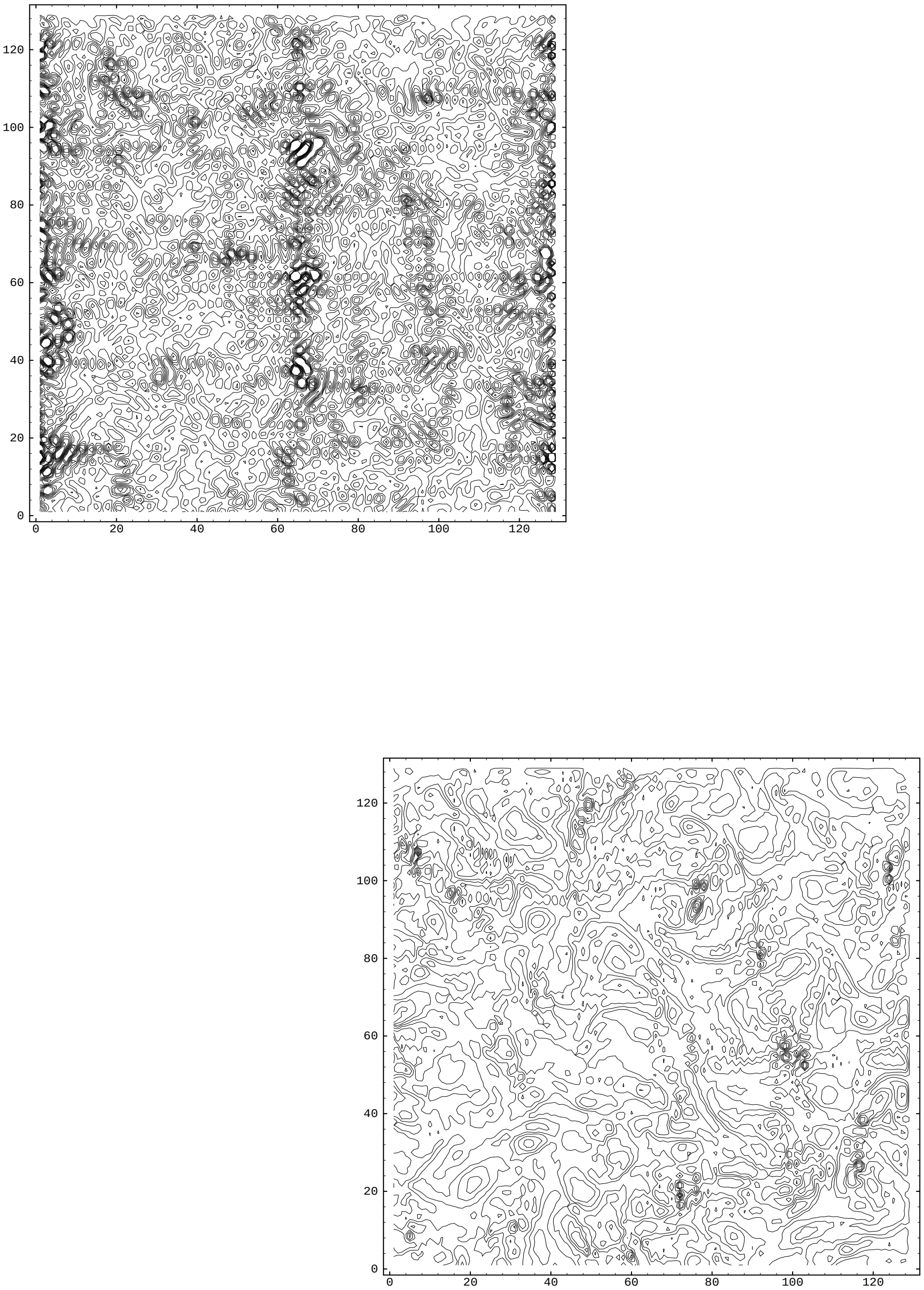

Figure 4: The vorticity at $t = 20$ from the initial conditions (3.1); $Re = 10^4$, resolution 64×64 (left plot) and 128×64 (right plot).

4 A Parallel Version of the Nonlinear Galerkin Method

The detailed description of the nonlinear Galerkin (NLG) method and a comprehensive list of references can be found in [4, 5, 9]. Here we develop a particular version of this method which can be implemented in combination with the Multidomain Local Fourier (MDLF) method.

Decomposition into large and small scales -

The NLG method starts with a decomposition of the velocity into its large and small scale components:

$$\mathbf{v} = \mathbf{y} + \mathbf{z}. \tag{4.1}$$

Based on a spectral decomposition, it was observed in [6] that most of the wavelengths (the small ones) carry only a small part of the total energy. Essential in the NLG method is a systematically differentiated treatment of the small and large scales. For example, short waves are not exactly integrated but their interaction with the large waves are taken into account in a simplified way. The aim of this approach is to recover the large eddies of the flow without explicitly computing all the scales of motion. The small eddies are in fact expressed as a nonlinear function of the large ones.

For periodic flows, $\mathbf{v}$ can be expanded into Fourier series:

$$\mathbf{v}(\mathbf{x},t) = \sum_{\mathbf{k}\in\mathcal{Z}^2} \hat{\mathbf{v}}(\mathbf{k},t)e^{i\mathbf{k}\cdot\mathbf{x}}, \qquad \mathbf{k} = (k_x, k_y). \tag{4.2}$$

By choosing a cut-off value $\mathbf{K} = (K_x, K_y)$, we can define $\mathbf{y}$ and $\mathbf{z}$ as follows:

$$\mathbf{y}(\mathbf{x},t) = \sum_{|k_x|\le K_x, |k_y|\le K_y} \hat{\mathbf{v}}(\mathbf{k},t)e^{i\mathbf{k}\cdot\mathbf{x}},$$

$$\mathbf{z}(\mathbf{x},t) = \left(\sum_{|k_x|>K_x, |k_y|\le K_y} + \sum_{k_x\in\mathcal{Z}, |k_y|>K_y} \right) \hat{\mathbf{v}}(\mathbf{k},t)e^{i\mathbf{k}\cdot\mathbf{x}}.$$

Our purpose is to apply the NLG method in multidomain regions. For a rectangular domain decomposed into parallel strips, we can expand $\mathbf{v}$ into Fourier series in the (periodic) y-direction, and into the Local Trigonometric Basis (LTB) in the x-direction (across the strips):

$$\mathbf{v}(\mathbf{x},t) = \sum_{k\in\mathcal{Z}} \sum_{l\in\mathcal{N}} \hat{\mathbf{v}}(k,l,t)e^{ik\cdot y}\phi_l(x), \tag{4.3}$$

where ϕ_l are the LTB functions (2.10). As before, we introduce a cut-off value $\mathbf{K} = (K_x, K_y)$ and then define:

$$\mathbf{y}(\mathbf{x},t) = \sum_{|k|\le K_x} \sum_{|l|\le K_y} \hat{\mathbf{v}}(k,l,t)e^{ik\cdot y}\phi_l(x), \tag{4.4}$$

and

$$\mathbf{z}(\mathbf{x},t) = \left(\sum_{|k|>K_x} \sum_{l\in\mathcal{N}} + \sum_{|k|\le K_x} \sum_{|l|>K_y} \right) \hat{\mathbf{v}}(k,l,t)e^{ik\cdot y}\phi_l(x). \tag{4.5}$$

Parallel implementation of the NLG method -

We incorporate the NLG method into the MDLF algorithm of section 2. Our NLG approach imposes a decomposition into large scales and small scales directly on the discretized system (2.5)-(2.7).

When solving the Poisson (2.6) and the Helmholtz (2.7) equations, we first apply the FFT in the y-direction. This results in a collection of one-dimensional ODEs (2.12) for the Fourier amplitudes $\hat{u}_k(x)$. Then we solve for the first $k \leq K$ amplitudes at each time step while the rest of the amplitudes, $k > K$, are kept "frozen" during several time iterations, that is $\mathbf{z}_K(t + p\Delta t) = \mathbf{z}_K(t)$. At the end of this frozen period, $\mathbf{z}$ is updated by integrating the corresponding equations. The outcome of this approach is that the small eddies are essentially "slaved" by the large ones.

The extent of the frozen period, $\tau = p\Delta t$, as well as the cut-off parameter, K, are found adaptively during the computation. For example, a possible criterion would be based on the magnitude of the ratios

$$\frac{|\mathbf{z}_K(t)|_2}{|\mathbf{y}_K(t)|_2} \quad \text{and} \quad \frac{||\mathbf{z}_K(t)||}{||\mathbf{y}_K(t)||}.$$

More sophisticated NLG strategies employ multi-level decomposition with adaptive determination of the cut-off values K_i (for details see [5]).

In the numerical examples given below, the choice of a particular NLG strategy is not significant since asymptotic (when time tends to infinity) shapes of the attractors are simple low-dimensional manifolds. However for more complicated examples, such as turbulent flows in pipes or channels, when small scales do not decay and affect the large eddies during a long time, an adaptive NLG strategy is expected to be of great importance.

5 Parallel Simulation of 2-D Turbulence with the NLG Method

In this section we present some numerical results on numerical simulations of turbulent flows by using the MDLF algorithm in combination with the NLG method. In the following examples the NLG algorithm is applied only in the periodic direction (along the strips) while in a cross-wise direction we use a complete expansion into the Local Trigonometric Basis without separation of scales. The domain decomposition is considered in the form of parallel strips, the number of subdomains (processors) is $P = 4$.

5.1 Decaying turbulence

Consider the decaying turbulence (the force $\mathbf{f} = 0$) starting with the initial conditions (3.1). The vorticity field is plotted in Fig. 5 at $t = 20$ when using the complete computation (updating the small scales at each time step) and the NLG method ($N_x = N_y = N = 128, k_x = k_y = N/2$, the cut-off value $K = k_y/2 = 32$). Remarkable that though the difference between these two plots is practically insignificant, the gain in the CPU time amounts to 50%, see Fig. 6. On the other hand, if we ignore completely the small scales $\mathbf{z}$, then the rate of energy dissipation will be not sufficient and the computation will eventually blow up, see Fig. 4.

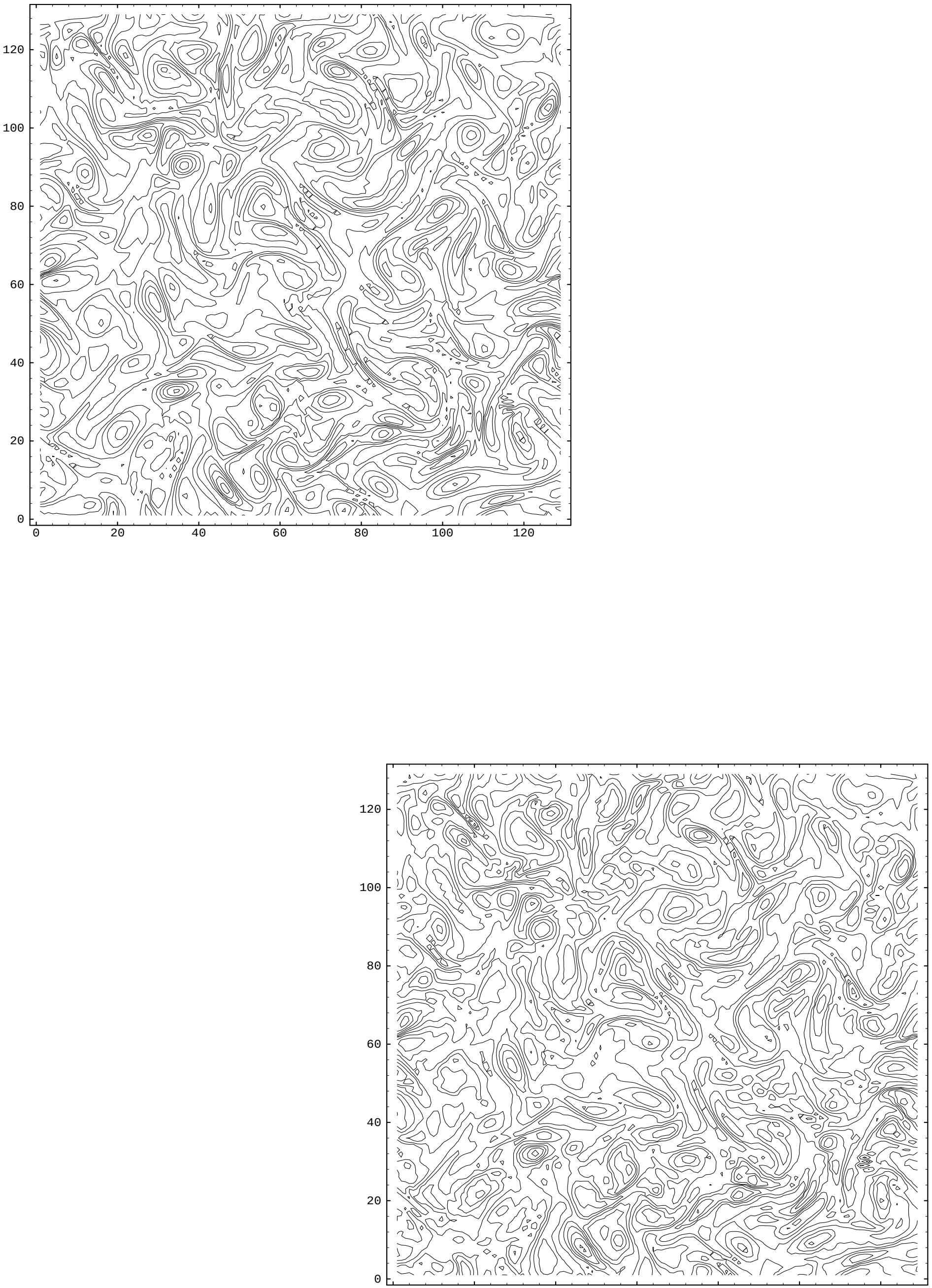

Figure 5: The vorticity at $t = 20$, $Re = 10^4$, resolution 128×128 obtained by using the conventional MDLF method (left plot) and the same method incorporating the NLG algorithm (right plot).

-bb-error = =

Figure 6: Saving of the CPU time by using the NLG method. 1- standard pseudo-spectral Fourier method. 2- NLG method with the same time step.

Another important effect of the NLG method is that it enables a stable time integration with a larger time step, see Fig. 7. The reason is that the Courant stability condition on the time step in such a case is determined by the large scales rather than by the small ones.

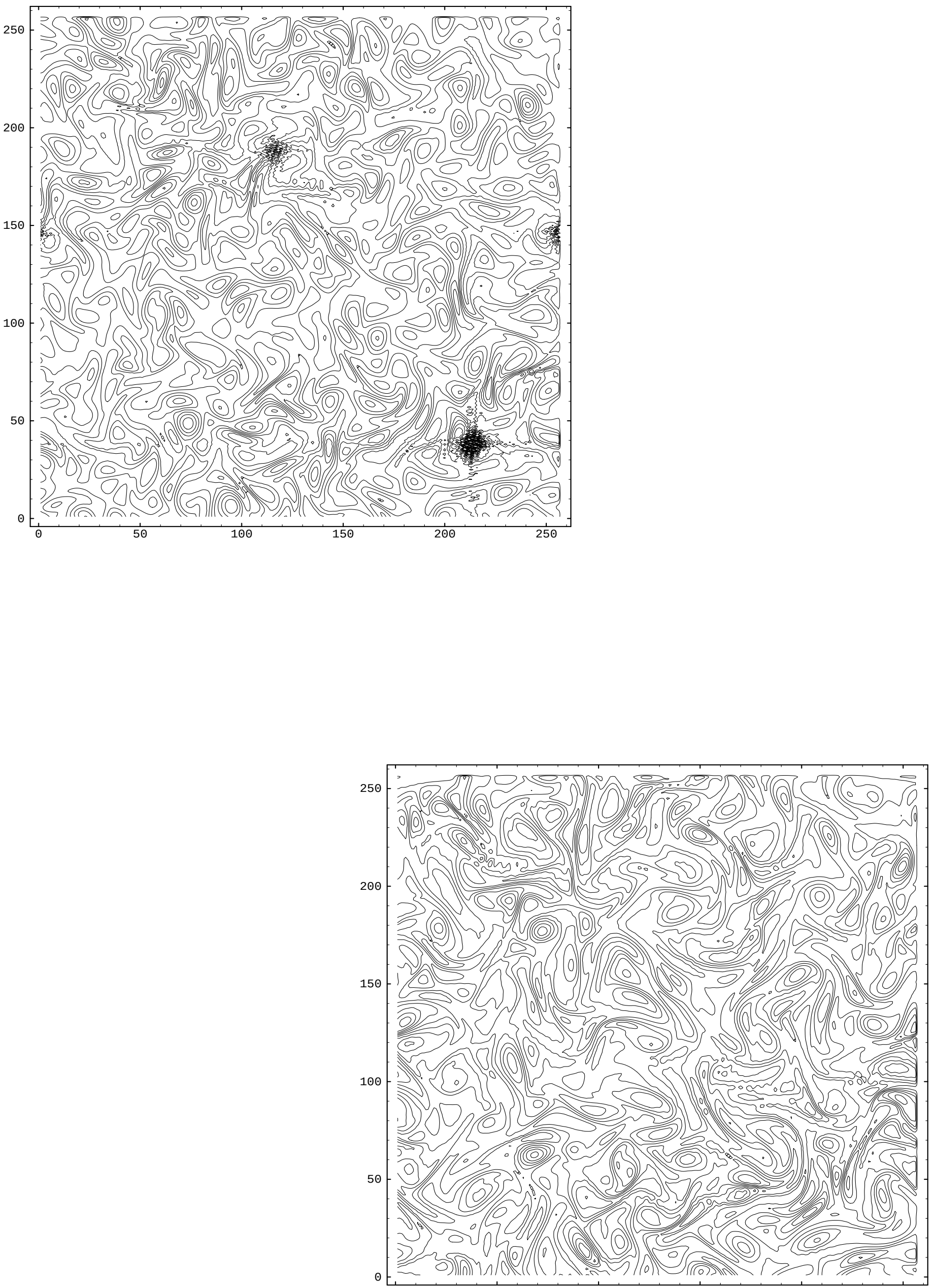

Figure 7: Effect of the NLG method on numerical stability.
Left plot: development of the numerical instability ($t = 7$, $Re = 10^4$, 128×256).
Right plot: stabilization of numerical integration by using the NLG method ($t = 20$).

5.2 Turbulence in the Presence of Forcing

It is expected that the presence of forcing will change the nature of the attractors and permit a quasi-steady long term behaviour of the turbulence. Initial experiments verify the applicability of our procedure. The following set of computations is performed with an external time-independent force, having only a few non-zero wavenumbers, see [4]:

$$\hat{\mathbf{f}}_{\mathbf{k}} = < \hat{f}_{1,\mathbf{k}}, \hat{f}_{2,\mathbf{k}} >$$

with

$$\begin{cases} |\hat{f}_{j,\mathbf{k}}| & = r \quad \text{if} \quad \mathbf{k} \in \mathbf{Z}^2, |k_x| + |k_y| = 3, \\ |\hat{f}_{j,\mathbf{k}}| & = 0 \qquad \text{otherwise} \end{cases} \tag{5.1}$$

where $j = 1, 2$. The Fourier coefficients of $\mathbf{f}$ are finally obtained by:

$$\hat{f}_{j,\mathbf{k}} = |\hat{f}_{j,\mathbf{k}}| e^{i\theta(\mathbf{k})}, \tag{5.2}$$

where phases $\theta(\mathbf{k}) \in [0, 2\pi]$ are randomly generated.

The initial conditions are chosen as in [4]:

$$|\hat{\zeta}(\mathbf{k})| = \begin{cases} \frac{C}{\sqrt{|\mathbf{k}| + (\sqrt{\nu}|\mathbf{k}|)^5}} & |\mathbf{k}| < 60 \\ 0 & |\mathbf{k}| \geq 60 \end{cases} \tag{5.3}$$

It was found that the presence and amplitude of the forcing profoundly changes the stability characteristics of the numerical computation. The following table illustrates the stability restriction on the time step for several amplitudes of the force and for two resolutions 128×128 and 256×256.

	amplitude of the force r				
Δt	0	0.01	0.02	0.05	0.1
0.001	s	s	s	s	n
0.002	s	s	s	n	n
0.005	s	s	s	n	n
0.01	s	s	s	n	n
0.02	s	s	n	n	n
0.05	s	n	n	n	n
0.1	s	n	n	n	n

	amplitude of the force r				
Δt	0	0.01	0.02	0.05	0.1
0.001	s	s	s	s	s
0.002	s	s	s	s	n
0.005	s	s	n	n	n
0.01	s	n	n	n	n
0.02	s	n	n	n	n
0.05	s	n	n	n	n
0.1	n	n	n	n	n

Table 2: Stability boundaries (s-stable, n- unstable) for resolutions 128^2 (left) and 256^2 (right)

The plots in Figs. 8-11 are obtained for the amplitude of the force $r = 0.02$.

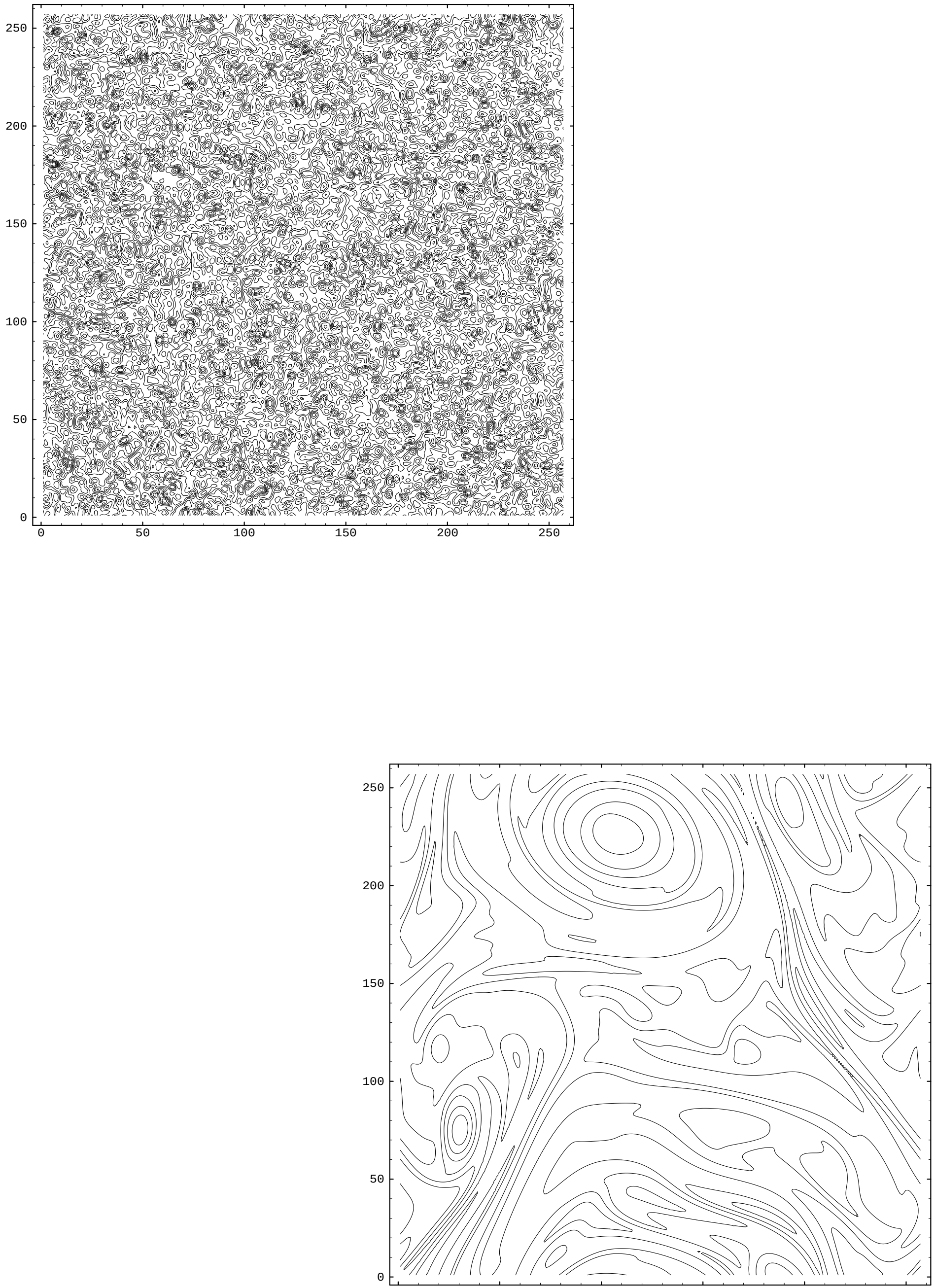

Figure 8: The vorticity at t=25 (right plot) from the initial field (5.2) (left plot) in the presence of the force (5.1)

The vorticity structure in Fig. 8 is similar to that obtained at the same parameters in [4] by using a serial code. The effect of the NLG method is demonstrated in Figs. 9-11.

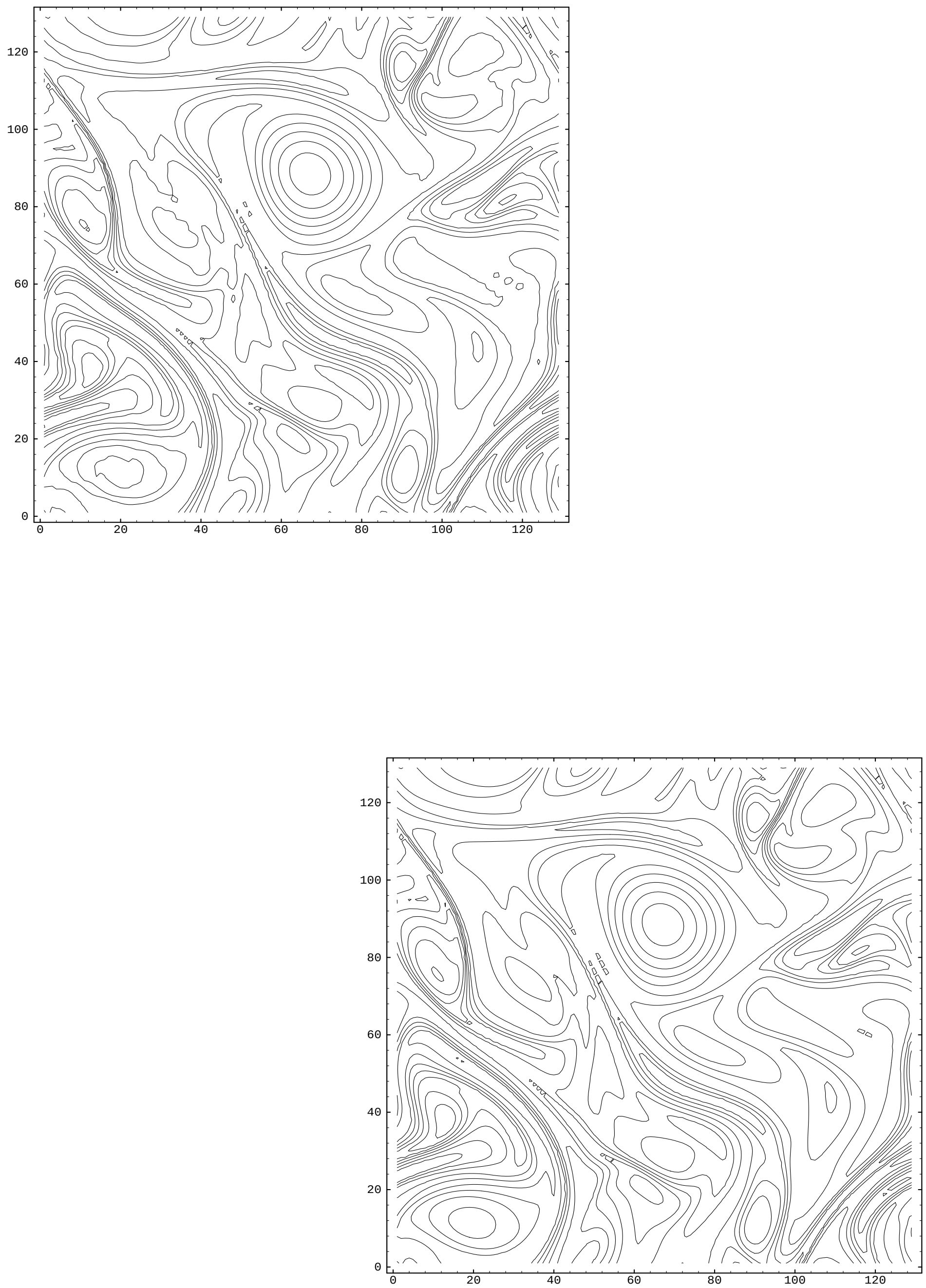

Figure 9: Comparison of contour plots of vorticity when using (left plot) and not using (right plot) the NLG method; $t = 30$, $Re = 10^3$, the mesh 128×128.

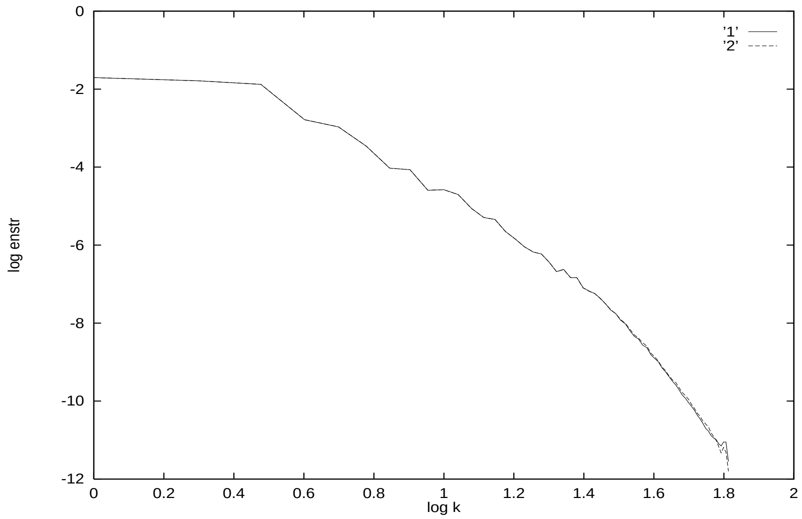

Figure 10: The enstrophy spectra at $t = 30$ when using (line 1) and not using (line 2) a NLG strategy.

-bb-error $=$ $=$

Figure 11: Saving of the CPU time by using the NLG method with the same time step.

6 *

Conclusions In this paper we demonstrate the feasibility of the parallel implementation of the Nonlinear Galerkin method. Incorporating the NLG approach in the present parallel algorithm may result in about 50% gain in the computing time while preserving good resolution of the small features in the flow. Another effect of the NLG method is that it allows the use of a larger time step (for a particular resolution) for the stable time integration.

Acknowledgement: We would like to thank Prof. Roger Temam for his guidance, technical support and consultation during the duration of this research. This research is a consequence of collaboration supported by the French AFIRST and the Israeli Ministry of Science.

References

[1] A. Averbuch, L. Ioffe, M. Israeli, L. Vozovoi, *Highly Scalable 2-D and 3-D Navier-Stokes Parallel Solver on MIMD Multiprocessors*, to appear in the J. of Supercomputing.

[2] G.L.Browning and H.-O.Kreiss, *Comparison of numerical methods for the calculation of two-dimensional turbulence*, Mathematics of Computation, **52**, No. 186 (1989), 369-388.

[3] R. Coifman, Y. Meyer, *Remarques sur l'analyse de Fourier à fenêtre, série*, C. R. Acad. Sci. Paris **312** (1991), 259-261.

[4] A. Debussche, T. Dubois, R.Temam, *The Nonlinear Galerkin Method: A multi-scale method applied to the simulation of homogeneous turbulent flows*, Theoretical and Computational Fluid Dynamics, 7,4(1995), 941-967..

[5] T. Dubois and R.Temam, *Separation of scales in turbulence using the Nonlinear Galerkin method*, Advances in Computational Fluid Dynamics, W.G.Habashi et M.Hafez ed., Gordon and Breach Publishers, 1993.

[6] C. Foias, O. Manley and R. Temam, *On the interaction of small and large eddies in two dimensional turbulent flows*, Math. Model. and Num. Anal., (M2AN) 22(1988), 93-114.

[7] M. Israeli, L. Vozovoi, A. Averbuch, *Spectral multi-domain technique with Local Fourier Basis*, J. Scientific Computing, **8**, (1993), 181-195.

[8] M. Israeli, L. Vozovoi, A. Averbuch, *Parallelizing implicit algorithms for time-dependent problems by parabolic domain decomposition*, J.Scientific Computing, **8**, (1993), 197-212.

[9] F.Jaubertau, C.Roisier, R.Temam, *The Nonlinear Galerkin method in computational fluid dynamics*, Applied Numercal Mathemtics, **6**, (1989/90), 361-370.

[10] G. E. Karniadakis, M. Israeli, S. A. Orszag, *High order splitting methods for the incompressible Navier-Stokes Equations*, J. Computational Physics, **97**, (1991), 414–443.

[11] R. Temam, *Inertial manifolds and multigrid methods*, SIAM J. Math. Anal., **21**, 154-178, (1990).

[12] L. Vozovoi, M. Israeli, A. Averbuch, *Spectral multi-domain technique with Local Fourier Basis II: decomposition into cells*, J. Scientific Computing, **9**, N 3, (1994), 311-326.