Physica Scripta. Vol. T116, 123–127, 2005

Parallel Simulations in Turbulent MHD

Daniel O. Gómez*[1], Pablo D. Mininni[2] and Pablo Dmitruk[3]

[1]Department of Physics, Pabellón I, C. Universitaria, 1428 Buenos Aires, Argentina
[2]Advanced Study Program, National Center for Atmospheric Research, P.O.Box 3000, Boulder CO 80307, USA
[3]Bartol Research Institute, University of Delaware, Newark, Delaware 19716, USA

Received July 9, 2004

PACS numbers: 4711+j, 4727Eq

Abstract

The large-scale dynamics of plasma flows can often be described within a fluidistic approximation known as one-fluid magnetohydrodynamics. Complex flows such as those corresponding to turbulent regimes are ubiquitous in laboratory plasmas and in astrophysics, because of their typically very large Reynolds numbers.

Numerical simulations have become a powerful tool for the study of complex plasma flows in recent years. The aim of the present paper is to introduce the reader to some of the standard numerical approximations used for the integration of the magnetohydrodynamic equations. In particular, we focus on pseudo-spectral methods and on how to develop parallel codes to speed up large Reynolds number simulations. We show the results arising from numerical simulations of astrophysical interest such as the development of turbulent flows in reduced magnetohydrodynamics and the generation of magnetic fields by dynamo mechanisms in three dimensional magnetohydrodynamics.

1. Introduction

Numerical simulations have become an important tool for fluid mechanics in recent years. The large-scale dynamics of plasma flows can often be described within a fluidistic approximation known as one-fluid magnetohydrodynamics. Complex flows such as those corresponding to turbulent regimes are ubiquitous in laboratory plasmas and in astrophysics, because of their typically very large Reynolds numbers.

The aim of the present paper is to introduce the reader to some of the standard numerical approximations to the integration of nonlinear partial differential evolution equations, such as those of magnetohydrodynamics (MHD). More specifically, we focus on the so called *Fourier-Galerkin* methods for problems with periodic boundary conditions, computationally more efficient than finite difference methods. To illustrate the use of these techniques, we show numerical results from simulations performed to study different astrophysical problems, such as the development of turbulence in externally driven reduce MHD (RMHD) or the generation of magnetic field by a turbulent dynamo in three dimensional MHD.

In section 2 we write down the MHD equations, Section 3 contains a summary of the numerical techniques used in our codes, while the development of parallel versions of these codes is discussed in section 4. In section 5 we show results from simulations of the reduced MHD equations, and in section 6 we present results from three dimensional MHD simulations. In section 7 we list the main conclusions of the present paper.

2. Theoretical framework

Magnetohydrodynamics is a reasonable theoretical framework to describe the macroscopic dynamics of plasmas, i.e. on typical lengthscales much larger than their collisional mean free paths and on timescales much longer than their collisional frequencies. If we also assume the flows to be incompressible, the MHD equations are

$$\partial_t \boldsymbol{B} = \nabla \times (\boldsymbol{U} \times \boldsymbol{B}) + \eta \nabla^2 \boldsymbol{B}, \tag{1}$$

$$\partial_t \boldsymbol{U} = -(\boldsymbol{U} \cdot \nabla)\boldsymbol{U} + (\nabla \times \boldsymbol{B}) \times \boldsymbol{B} - \nabla p + \nu \nabla^2 \boldsymbol{U}, \tag{2}$$

$$\nabla \cdot \boldsymbol{B} = 0, \qquad \nabla \cdot \boldsymbol{U} = 0. \tag{3}$$

The magnetic field in these equations is in velocity units, i.e. $\boldsymbol{B} = B[\mathrm{Gauss}]/\sqrt{4\pi\rho}$, and ρ is the (constant) mass density of the plasma. The dissipation coefficients η and ν are the resistivity and kinematic viscosity, respectively. We can transform Eqs. (1)–(3) into a dimensionless set of equations by scaling the velocity and magnetic fields to a characteristic speed U_0, and lengths to a typical length L_0. Two dimensionless numbers will naturally arise to measure the relative importance of nonlinearities compared to dissipation. Namely, the well known Reynolds number, $R = U_0 L_0/\nu$, and the magnetic Reynolds number, $R_m = U_0 L_0/\eta$.

Many astrophysical flows are characterized by Reynolds numbers much larger than unity ($R \sim R_m \sim 10^{10-12}$) for which nonlinearities are the dominant physical process. The effect of nonlinear terms is to stochastically redistribute excitations from one wavenumber to another, while excitations at sufficiently large wavenumbers decay by dissipative effects. Therefore, a net flow of energy in Fourier space is established, known as the energy cascade. Kolmogorov 1941 [1], following essentially dimensional arguments, has shown that an externally driven three dimensional incompressible fluid displays a stationary energy spectrum given by $E_k \propto k^{-5/3}$. Kolmogorov's ideas are usually known as *cascade theory* and have been applied to a number of turbulent systems including two and three-dimensional MHD turbulence [2]. The power spectra predicted by cascade theory for the energy inertial range, have in many cases been confirmed by experiments and numerical simulations.

3. Numerical recipes

In this section we describe several basic concepts about the numerical simulation of MHD flows using spectral methods (for a general overview on spectral methods applied to fluid simulations see [3], and also [4], [5] and [6]). Hereafter we will focus on *Galerkin-Fourier* methods, which are the proper ones for problems displaying periodic boundary conditions. To illustrate the method, let us consider Burgers equation, which contains all

*Also at IAFE (CONICET, UBA), C. Universitaria, 1428 Buenos Aires, Argentina. *Email address*: dgomez@df.uba.ar (Daniel O. Gómez). *URL*: http://astro.df.uba.ar (Daniel O. Gómez).

the ingredients (spatial and temporal derivatives, quadratic nonlinearity and dissipation) of the more complex HD and MHD problems in two and three dimensions. Burgers equation describes the evolution of a 1D velocity field $u = u(x, t)$,

$$\partial_t u + u\partial_x u = \nu\partial_{xx} u. \tag{4}$$

This equation is supplemented with the initial condition $u(x, 0) = u_0(x)$ and periodic boundary conditions for x in the interval $[0, 2\pi)$. This allows for a Fourier expansion of $u(x, t)$, which is numerically approximated by the truncation

$$u^N(x, t) = \sum_{k=-N/2}^{N/2} u_k(t)\mathrm{e}^{\mathrm{i}kx}. \tag{5}$$

Spectral methods demand that the error introduced by this truncation, has zero projection on the subspace generated by $\{\mathrm{e}^{\mathrm{i}kx}, k = -N/2 + 1, \ldots, N/2\}$. As a result, a PDE like Eqn. (4) is replaced by a set of coupled ODEs for the Fourier coefficients $u_k(t)$ (for $k = -N/2 + 1, \ldots, N/2$)

$$\partial_t u_k = -(u\partial_x u)_k - \nu k^2 u_k, \qquad (u\partial_x u)_k = \sum_{l+m=k} i\, m u_l u_m. \tag{6}$$

For continuous, infinitely differentiable and periodic functions, the numerical convergence is exponentially fast [3]. Before Eqs. (6) can be used as an approximation to Eqn. (4), we must note that the nonlinear term involves $O(N^2)$ floating point operations. By comparison, the evaluation of $u\partial_x u$ in physical space (by finite differences) would require $O(N)$ floating point operations. To overcome this practical difficulty, a "fast Fourier transform" (FFT) [7] is used, which reduces the computational load to only $O(N \log N)$ operations. This strategy of computing spatial derivatives in physical space and non-linear terms in physical space is known as "pseudospectral", and combines the advantages of computational efficiency and high precision.

There is a spurious effect known as "aliasing", which stems from the fact that the functions $\mathrm{e}^{\mathrm{i}kx}$ and $\mathrm{e}^{\mathrm{i}(k+Nm)x}$ are indistinguishable when evaluated on the set $\{x_j = 2\pi j/N, j = 0, \ldots, N-1\}$. If we restrict our numerical evaluation to the discrete set of wavenumbers $k = -N/2 + 1, \ldots, N/2$, the harmonic modes with wavenumbers equal to $k + Nm$ do not take part in the computation of any linear term. However, the nonlinear terms cause the coupling of Fourier modes, and therefore harmonic modes will produce aliasing. To overcome this difficulty, the standard "two-thirds rule" will be applied (see [3] for details). This rule consists in forcing all modes with $|k| > N/3$ (i.e. $2/3$ of all Fourier modes) to have zero amplitude, which guarantees that the spurious coupling with harmonic modes is exactly zero.

For the time integration of Eqs. (6), we use a second order Runge-Kutta scheme. Let us write down this set of equations with the following compact notation.

$$\frac{\mathrm{d}U}{\mathrm{d}t} = F(U, t) \tag{7}$$

where U is a vector containing the Fourier coefficients. To numerically integrate this equation (see details in [3]) in discrete time steps of size Δt we first advance half a step

$$U^{t+\frac{1}{2}} = U^t + \frac{\Delta t}{2} F(U^t, t) \tag{8}$$

and then use $U^{t+\frac{1}{2}}$ to evaluate the right hand side of Eqn. (7) and advance a full step

$$U^{t+1} = U^t + \Delta t F\big(U^{t+\frac{1}{2}}, t + \tfrac{1}{2}\big). \tag{9}$$

Higher order Runge-Kutta schemes can provide more accurate results in exchange of a larger consumption of CPU time and larger use of RAM memory. The second order scheme proves to be sufficiently accurate for our purposes. Predictor-corrector schemes are also regularly used for hydrodynamic simulations. In "second order predictor-corrector", both U^t and U^{t-1} are used to extrapolate the solution one step advanced, and then correct this extrapolation using information of the derivative at the new point. We also used this scheme in some of our simulations, treating the linear (dissipative) terms implicitly and the nonlinear terms explicitly (see also [3], [8]). Both methods are accurate to second order in Δt.

For nonlinear partial differential equations as the ones being considered, there are no clear rules to guarantee the numerical stability of a simulation, and therefore there are no recipes indicating how small Δt ought to be. Nonetheless, an estimate originated in the CFL (Courant-Friedrichs-Lewy) condition for advection [3] provides a reasonable upper bound. For an advection equation such as $\partial_t u \sim u\partial_x u$, integrated with a spatial grid size Δx, the required time step is $\Delta t \leq \Delta x/u_0$, where u_0 is a typical velocity.

4. Parallel simulations

The parallelization of the pseudo-spectral codes described above, has been performed using the MPI (Message Passing Interface) library. MPI is currently a standard implementation to parallelize high performance numerical codes on both shared and distributed memory platforms. The exchange of information between processors in a cluster is performed through MPI.

In sections 5 and 6 we show two applications of our codes, which also correspond to very different examples for parallelization. Reduced MHD (in section 5) is more amenable for parallelization, since it only involves first order derivatives along the z-direction, which is the direction of the main magnetic field. To compute first order derivatives by finite differences to second order accuracy, say at the grid point $z = i\Delta z$, we only need to have the state of the system at the neighboring points $z = (i \pm 1)\,\Delta z$. If for simplicity we assume that the simulation at any gridpoint $z = i\Delta z$ $(i = 1, \ldots, N_z)$ is being carried out at a different processor, we only need to exchange information between "neighboring" nodes to compute these derivatives. The communication load is by comparison much smaller than the numerical load being handled by each processor, and therefore the parallel version runing in a cluster of N_z nodes runs almost N_z times faster.

On the other hand, in three dimensional simulations with periodic boundary conditions in the three components (see section 6), we basically need a parallel version of the FFT routine. Based on the FFTW open source routine [7], we implemented a parallel version that works as follows. The spatial distribution of each scalar field and component of vector field can be visualized as a cubic array of N^3 real numbers. Let us assume that each node handles a slice of $N \times N \times M$ numbers, where M is the linear number of grid points (N) divided by the number of processors. The Fourier transforms in the directions indicated by arrows in Fig. 1 (left) are performed locally in each node, and stored in a cube of size $(N/2 + 1) \times N \times M$. To compute the Fourier transform in the remaining direction, requires an all-to-all node communication step, to transpose the full matrix and split it in slices of $P \times N \times N$. The number P is now equal to $N/2 + 1$ divided by the number of processors. The Fourier transform in the remaining direction (indicated by the arrow on the right frame of Fig. 1), can now be performed locally in each node [10].

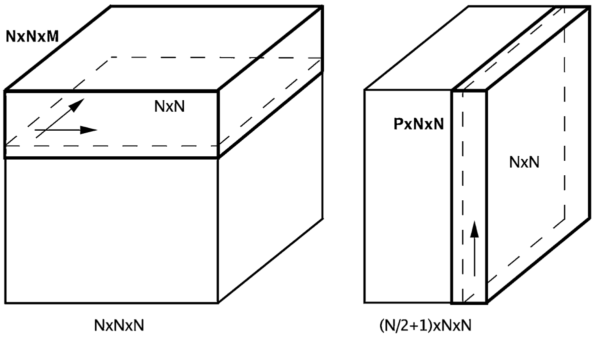

Fig. 1. Left: box of any of the unknowns in real space for an N^3 simulation. Each node handles a slice of $N \times N \times M$. Fourier transforms in the directions indicated by arrows are performed locally in each node, and stored in a cube of size $(N/2+1) \times N \times M$. Right: An all-to-all node communication step is used to transpose the full matrix and split it in slices of $P \times N \times N$, so that the Fourier transform in the remaining direction (see arrow) can be performed locally in each node.

This numerical strategy is quite efficient (see also [9]), although the exchange of information between nodes is much bigger than for the RMHD case. The efficiency in exchanging communication between nodes is measured by the speed-up factor, which is simply the ratio between the time it takes a code to run on a single processor and the run time in an multi-processor cluster. In the next two sections we quantitatively show the speed-up for the two applications considered.

5. Turbulence in reduced MHD

In this section, we focus on the dynamics of a relatively homogeneous bundle of magnetic field lines, which are described by the reduced MHD approximation [11]. A typical example of such a system is a magnetic loop in the solar corona, whose footpoints are deeply rooted into the solar photosphere, thus generating magnetic stresses in the coronal portion of the loop. Let us consider a loop with length L and cross section $2\pi l_p \times 2\pi l_p$, where l_p is the lengthscale of typical photospheric motions. The main magnetic field $\boldsymbol{B}_0$ is assumed to be uniform and parallel to the axis of the loop (the z axis). The planes at $z = 0$ and $z = L$ correspond to the loop footpoints at the photosphere.

The very high electric conductivity allows photospheric motions to easily drive magnetic stresses in the corona [12]. The field lines twist and bend due to these motions and this generates transverse components of velocity ($\boldsymbol{u}$) and magnetic field ($\boldsymbol{b}$), giving $\boldsymbol{b} = \nabla \times (a\hat{z})$ and $\boldsymbol{u} = \nabla \times (\psi\hat{z})$. The reduced MHD equations for the stream function ψ and the vector potential a are [11]:

$$\partial_t a = B_0 \partial_z \psi + [\psi, a] + \eta \nabla^2 a, \tag{10}$$

$$\partial_t \omega = B_0 \partial_z j + [\psi, w] - [a, j] + \nu \nabla^2 w, \tag{11}$$

where $w = -\nabla^2 \psi$ is the z-component of the fluid vorticity and $j = -\nabla^2 a$ is the z-component of the current density. The brackets $[A, B]$ are the standard Poisson brackets.

We numerically integrated Eqs. (10)–(11). To this end, ψ and a are expanded in Fourier modes in each (x, y) plane ($0 \leq x, y \leq 2\pi$ and $0 \leq z \leq 1$), as mentioned in Section 3. The corresponding Fourier coefficients $\psi_{\boldsymbol{k}}(z, t)$ and $a_{\boldsymbol{k}}(z, t)$ are evolved in time using a semi-implicit scheme: linear terms are treated in a fully implicit fashion, while nonlinear terms are evolved using a second order Runge-Kutta scheme (see Section 3).

To compute z-derivatives we use a standard method of finite differences in a staggered regular grid of $N_z + 1$ points. The stream function is computed on points $z_i = i/N_z$ ($i = 0, \ldots, N_z$), while the magnetic flux function is computed on $z_{i+1/2} = (i + 1/2)/N_z$ ($i = 0, \ldots, N_z - 1$). Boundary conditions for the stream function ψ are given at the plates $z = 0$ and $z = 1$. Therefore Eqn. (11) is not integrated on these planes, but it is evolved in time in all the internal gridpoints $z_i = i/N_z$ ($i = 1, \ldots, N_z - 1$).

A Linux cluster is used to run the parallel RMHD code. Since finite-differences are used in the z-direction, while a pseudospectral method is employed for the transverse directions, an efficient parallelization is achieved by performing the transverse gradients locally in each machine. Communication between nodes is only required to perform the z-derivatives. Figure 2 shows the speed-up of a parallel RMHD simulation, measured as the time ratio of the simulation in a single node by the simulation in a cluster of nodes, as a function of the number of nodes. We see that the speed up remains very close to the ideal situation, which is equal to the number of processors. RMHD is therefore an ideal plasma configuration for parallel simulations in commodity type of clusters (Beowulf).

In Figure 3 we show the magnetic and kinetic energy vs. time (left) for a simulation involving $1024 \times 1024 \times 128$ gridpoints, with a stationary velocity field imposed at $z = 1$ to induce a turbulent regime. We observe that both the energy and the energy dissipation rate (right) reach a statistically stationary regime. In Fig. 4 (left) we see the corresponding power spectra for the magnetic and kinetic energy, both of which display a broadband, power-law behavior. In Fig. 4 (right) we observe the power

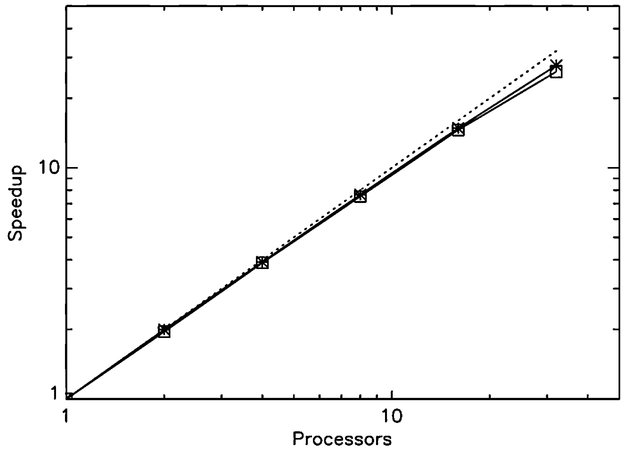

Fig. 2. Speed-up (time of a sequential run/time of a parallel run) of the RMHD code in a cluster of 32 PCs. Square correspond to a resolution $128 \times 128 \times 128$, asteriscs correspond to a resolution $512 \times 512 \times 128$. The dotted line is the ideal speed-up (equal to the number of processors).

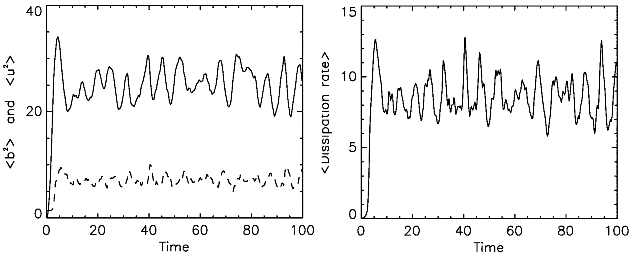

Fig. 3. Left: magnetic energy (full) and kinetic energy (dashed) as a function of time for an RMHD run with $R = S = 800$ and resolution $1024 \times 1024 \times 128$. Right: dissipation rate vs. time.

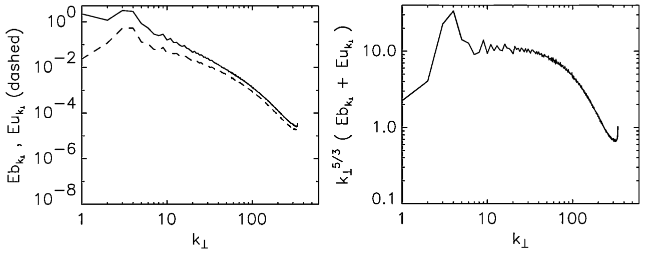

Fig. 4. Left: perpendicular magnetic energy (full) and kinetic energy (dashed) spectra for an RMHD run with $R = S = 800$ and resolution $1024 \times 1024 \times 128$. Right: total energy spectra compensated by $k^{5/3}$.

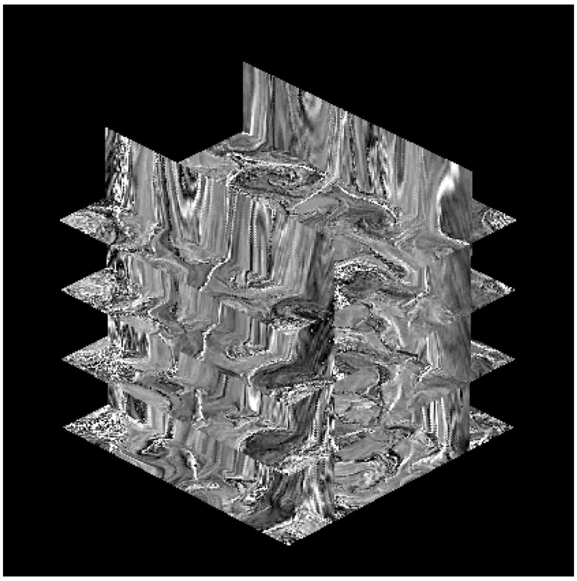

Fig. 5. Electric current density $j(x, y, z)$ in different cross sections of the RMHD simulation box, corresponding to a stationary turbulent regime.

spectrum of the total energy, compensated by a factor $k^{5/3}$, to show that it corresponds to a Kolmogorov slope in the inertial range of the energy cascade [13]. In Figure 5 we show the spatial distribution of the electric current density, corresponding to a stationary turbulent regime. We observe that the electric current flows both upward (white) and downward (black) along very elongated structures (see vertical slices). There is a very complex distribution of these structures, as can be seen in the horizontal slices. We speculate that the Joule dissipation originating in these fine scale structures, provides a promising mechanism to heat coronal loops (see [14] and references therein).

6. Three dimensional MHD turbulence

Pseudospectral methods can also be used to perform 3D simulations of incompressible flows for a large number of applications. As an illustrative example, in this section we focus on helical MHD flows displaying dynamo activity.

Externally driven MHD flows are expected to relax to a stationary regime characterized by an energy dissipation rate ϵ and a Kolmogorov energy power spectrum $E_k = C_K \epsilon^{2/3} k^{-5/3}$, where C_K is the Kolmogorov constant. This expression arises from the ideal invariance of the total energy (i.e. kinetic plus magnetic) of the system. The content of kinetic helicity $H = \frac{1}{2} \int d^3 r \boldsymbol{U} \cdot \nabla \times \boldsymbol{U}$ in flow motions is particularly relevant for dynamo mechanisms, i.e. for the generation of magnetic fields in plasma flows. According to mean field theories, from where the so called α-effect is derived [15], the growth rate of macroscopic magnetic fields is proportional to the content of kinetic helicity in microscopic flows. We performed 3D MHD simulations to quantitatively study turbulent dynamo mechanisms (see [16] and references therein). These simulations start from a purely hydrodynamic stage, applying an external mechanical force until a turbulent stationary regime is reached. In Figure 6 (left) we show the speed-up for hydrodynamic simulations (left), both for a modest resolution of 64^3 gridpoints and for a slightly better resolution of 128^3. For a small number of nodes, the speed up in both cases is close to ideal, since most of the numerical load is taken by the processors, with comparatively negligible exchange of information. As the number of processors increase, each processor handles a smaller load and communication becomes more important. We observe that the 64^3 case departs from the ideal speed-up at about 32 processors, while the 128^3 case remains close to ideal up to 96 nodes. Fig. 6 (right) directly shows the real time for a single time step, as a function of the number of processors for clusters with different connectivity speeds (1 Gbps and 100 Mbps), for comparison.

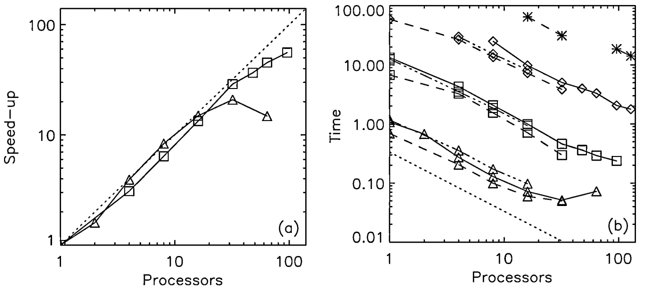

Fig. 6. Left: Speed-up for a pseudo-spectral hydrodynamics simulations of size 64^3 (triangles) and 128^3 (squares), performed in an IBM SP/3 cluster. The dotted line indicates the ideal speed-up. Right: Time (in seconds) for a single timestep for 64^3 (triangles), 128^3 (squares), 256^3 (diamonds) and 512^3 (stars) hydrodynamic runs. Simulations have been performed in dual nodes with a 1 Gbps network (dotted line) and in a Linux cluster with a 100 Mbps network (dash-dotted). The slope of the dotted line corresponds to ideal speed-up.

To generate magnetic fields in a turbulent flow, we proceed as follows. Once a stationary turbulent regime is reached, small magnetic field fluctuations are added at small scales (a magnetic seed), and the subsequent evolution of the velocity and magnetic fields is studied. After an initial stage during which the magnetic field intensity grows exponentially fast, a saturation level is reached, corresponding to the approximate equipartition between kinetic and magnetic energy. The power spectrum for the kinetic and magnetic energies are displayed in Fig. 7. The total energy power spectrum (thick trace) is consistent with a Kolmogorov law, with a value for the Kolmogorov constant $C_K \approx 1.39$.

Fig. 8 shows the spatial distribution of the perpendicular component of the magnetic field on a slice, at two different times. At the beginning of the simulation we observe the fine scale structure of the magnetic seed (left). At much longer times, when the stationary regime is reached, we observe a much longer correlation length (right). In summary, these three dimensional simulations show that helical flows are quite efficient in generating large scale magnetic fields.

7. Conclusions

The large scale dynamics of astrophysical flows, can often be described within the theoretical framework of magnetohydrodynamics. Important astrophysical problems, such as the generation of magnetic fields by dynamo mechanisms, the development of

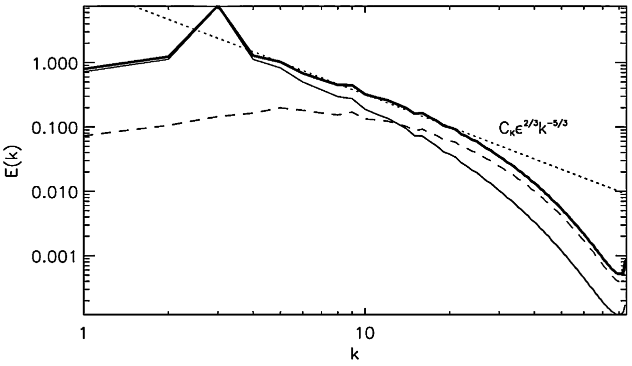

Fig. 7. Power spectra of total energy (thick), kinetic energy (thin), and magnetic energy (dashed) for a helical 3D MHD simulation of 256^3 grid points. The external helical forcing is centered at $k = 3$. The straight line corresponds to a slope $-5/3$ and a Kolmogorov constant $C_K = 1.39$.

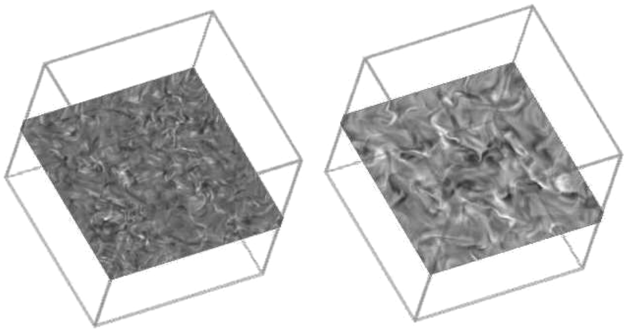

Fig. 8. Slice of a 256^3 simulation displaying the perpendicular component of the magnetic field. Left: shortly after the run starts. Right: when the stationary regime is reached.

turbulent regimes or the occurrence of magnetic reconnection, fall within this category.

In the present paper we discuss numerical approximations to the integration of the MHD equations. In particular, we briefly describe the *Fourier-Galerkin* method for problems with periodic boundary conditions. We use this method in a number of codes that integrate the MHD equations in different geometric configurations. For plasmas permeated by an approximately uniform magnetic field, the corresponding evolution of the velocity and magnetic field is described by the reduced MHD approximation. Examples of this configuration are the coronal holes or the magnetic loops in active regions of the solar corona. We show the development of a stationary turbulent regime when coronal loops are being externally driven by large-scale motions applied at their footpoints.

In section 6 we show 3D simulations of helical MHD flows, to study the role of kinetic helicity in generating magnetic fields. We confirm that the content of kinetic helicity in small-scale flows is essential to drive a turbulent dynamo, i.e. to generate large-scale magnetic fields. In summary, we find that numerical simulations are a powerful and promising tool for the study of large-scale plasma flows.

Acknowledgments

We gratefully acknowledge funding from the Agencia Nacional de Promoción de la Ciencia y la Tecnología (ANPCyT) through grant PICT2000 03-9483. PDM acknowledges computer time provided by equipment purchased under NSF ARI Grant CDA-9601817 and NSF sponsorship of NCAR. PD acknowledges partial support from grant NSF ATM-9977692. DOG is a member of the Carrera del Investigador from CONICET.

References

1. Kolmogorov, A. N., C. R. Acad. Sci. URSS **30**, 301 (1941).
2. Montgomery, D., in “Solar wind V” (ed. M. Neugebauer), NASA Conf. Publ. 2280 (1983), p. 107.
3. Canuto, C., Hussaini, M. Y., Quarteroni, A. and Zang, T. A., “Spectral Methods in Fluid Dynamics”, (Springer, New York, 1988).
4. Patterson, G. and Orszag, S. A., Phys. Fluids **14**, 2353 (1971).
5. Gottlieb, D. and Orszag, S. A., “Numerical Analysis of Spectral Methods: Theory and Application”, (SIAM, Philadelphia, 1977).
6. Gottlieb, D., Hussaini, M. Y. and Orszag, S. A., “Spectral Methods for Partial Differential Equations”, (SIAM, Philadelphia, 1984).
7. Frigo, M. and Johnson, S. G., in “Proc. IEEE Intl. Conf. Acoustics Speech and Signal Processing”, **3**, 1381 (1998).
8. Fletcher, C. A. J., “Computational Techniques for Fluid Dynamics” (2nd ed.), (Springer-Verlag, Heidelbery, 1991).
9. Dmitruk, P. *et al.*, Parallel Computing **27**, 1921 (2001).
10. Swarztrauber, P., Parallel Computing **5**, 197 (1987).
11. Strauss, H., Phys. Fluids **19**, 134 (1976).
12. Parker, E. N., Astrophys. J. **174**, 499 (1972).
13. Dmitruk, P., Gómez, D. and Matthaeus, W., Phys. Plasmas **10**, 358 (2003).
14. Dmitruk, P. and Gómez, D., Astrophys. J. Lett. **527**, L63 (1999).
15. Moffat, H. K., “Magnetic field generation in electrically conducting fluids”, (Cambridge Univ. Press, Cambridge, UK, 1978).
16. Mininni, P. D., Gómez, D. O. and Mahajan, S. M., Astrophys. J. **587**, 472 (2003).