

## MHD Turbulence Studies using Lattice Boltzmann Algorithms

G. Vahala<sup>1,\*</sup>, B. Keating<sup>1,2</sup>, M. Soe<sup>3</sup>, J. Yepez<sup>4</sup>, L. Vahala<sup>5</sup>, J. Carter<sup>6</sup> and S. Ziegeler<sup>7</sup>

<sup>1</sup> Department of Physics, William & Mary, Williamsburg, VA 23185, USA.

<sup>2</sup> Department of Ocean & Resources Engineering, University of Hawaii at Manoa, Honolulu, HI 96822, USA.

<sup>3</sup> Department of Mathematics and Physics, Rogers State University, Claremore, OK 74017, USA.

<sup>4</sup> Air Force Research Laboratory, Hanscom AFB, Bedford, MA 01731, USA.

<sup>5</sup> Department of Electrical & Computer Engineering, Old Dominion University, Norfolk, VA 23529, USA.

<sup>6</sup> NERSC Division, Lawrence Berkeley National Laboratory, Berkeley, CA 94720, USA.

<sup>7</sup> High Performance Computing Modernization Program, Mississippi State University, MS 39762, USA.

Received 31 October 2007; Accepted (in revised version) 4 December 2007

Available online 21 April 2008

---

**Abstract.** Three dimensional free-decaying MHD turbulence is simulated by lattice Boltzmann methods on a spatial grid of  $8000^3$  for low and high magnetic Prandtl number. It is verified that  $\nabla \cdot B = 0$  is automatically maintained to machine accuracy throughout the simulation. Isosurfaces of vorticity and current show the persistence of many large scale structures (both magnetic and velocity) for long times — unlike the velocity isosurfaces of Navier-Stokes turbulence.

PACS: 52.65.Kj, 52.35.Ra, 47.27.-i

**Key words:** Turbulence, lattice Boltzmann method, entropy, MHD.

---

## 1 Introduction

Here we examine free decaying 3D magnetohydrodynamics (MHD) by a mesoscopic algorithm that, unlike standard computational fluid dynamic (CFD) algorithms, is amenable

---

\*Corresponding author. *Email addresses:* gvahala@gmail.com (G. Vahala), brkeats@gmail.com (B. Keating), msoe.rsu@gmail.com (M. Soe), jeffrey.yepez@gmail.com (J. Yepez), lvahala@odu.edu (L. Vahala), jtcarter@lbl.gov (J. Carter), sean.ziegeler@nrlssc.navy.mil (S. Ziegeler)

to massive parallelization [1]. Indeed, our lattice Boltzmann (LB) code [1] has had a sustained performance of 26.25 TFlops/s on 4800 PEs of the *Earth Simulator* — i.e., 67% of peak and outputting 0.25 TB of data. Moreover, the  $\nabla \cdot \mathbf{B} = 0$  constraint is automatically enforced, thus side-stepping the need for divergence cleaning. Our work is a generalization of the seminal 2D LB-MHD algorithm of Dellar [2].

The basic idea behind the LB method [3,4] is to project the desired nonlinear macroscopic system into a higher dimensional phase space with the resulting kinetic system simpler to solve and readily parallelized. The difficult nonlinear convective derivatives  $\mathbf{u} \cdot \nabla \mathbf{u}$ ,  $\mathbf{u} \cdot \nabla \mathbf{B}$ ,  $\dots$  (where  $\mathbf{u}$  is the fluid velocity and  $\mathbf{B}$  the magnetic field) of CFD are now replaced by simple linear advection (a shift operation) and local collisional relaxation in phase space. On performing the Chapman-Enskog long-time long-wavelength asymptotics [2–4] on the discretized LB system, one recovers the MHD equations to leading order in the Knudsen number and thus relating the MHD transport coefficients to the relaxation parameters in the BGK collision operators of LB. The essential point is that non-local macroscopic gradients, like the mean strain rate or  $\nabla \cdot \mathbf{B}$ , are computed at the mesoscopic LB level by simple local moments of the distribution functions. To recover [2–4] the Navier-Stokes equation, one need only introduce a scalar distribution function  $f(\mathbf{x}, \boldsymbol{\zeta}, t)$  whose zeroth moment yields the density and first moment yields the momentum. The importance of Dellar’s work [2] was his introduction of a vector distribution function  $\mathbf{g}(\mathbf{x}, \boldsymbol{\zeta}, t)$  whose zeroth moment defines the magnetic field  $\mathbf{B}$ .

One minimizes the computational memory requirements resulting from the transformation from  $(\mathbf{x}, t)$ - to  $(\mathbf{x}, \boldsymbol{\zeta}, t)$ -space by a clever choice of discretization of  $\boldsymbol{\zeta}$ -space. In particular, it has been shown [3,4] that one can recover the 3D Navier-Stokes equation with a 15-bit discretization of  $\boldsymbol{\zeta}$ -space. One must also consider the numerical stability of LB — especially as one pushes to smaller and smaller transport coefficients — since LB is an explicit, second order accurate scheme. In its simplest formulation [3,4], there are no constraints imposed on the discretized velocity distribution function  $f_\alpha(\mathbf{x}, t)$  to maintain its positive definiteness throughout the simulation. Recently this problem has been successfully addressed for the Navier-Stokes equation [5–11] by imposing an entropy constraint on the discretized  $f_\alpha(\mathbf{x}, t)$  to enforce positive-definiteness. This has resulted in an entropic lattice Boltzmann (ELB) scheme that is unconditionally stable. An outstanding problem is whether a similar ELB scheme can be devised to LB MHD.

In Section 2, we briefly introduce the LB and ELB schemes for Navier-Stokes turbulence and introduce the lattice discretization of  $\boldsymbol{\zeta}$ -space by a 15-, 19- or 27-velocities at each spatial node. We then introduce our 3D LB MHD representation. The parallelization and performance our LB schemes on various supercomputer architectures is discussed in Section 3. In Section 4 we first present some of our basically fully resolved ELB simulations for Navier-Stokes turbulence on a  $1600 \times 1600 \times 1600$  spatial grid at a Reynolds number of 25000. These simulations clearly indicate intermittency [12] in the turbulence by the deviation of the energy spectrum from the  $k^{-5/3}$  Kolmogorov spectrum. In Section 4.2, we present LB-MHD simulations on a  $1800 \times 1800 \times 1800$  spatial grid for magnetic Prandtl number  $\text{Pr} = 0.3$  and  $\text{Pr} = 3.0$ , where  $\text{Pr}$  is the ratio of the viscosity

to resistivity. The vorticity and current isosurfaces are presented for the high Pr case and show the perseverance of large scale magnetic and velocity structures.  $\nabla \cdot \mathbf{B} = 0$  is maintained to machine accuracy  $\mathcal{O}(10^{-15})$ . We also consider the magnetic field correlations and find that they are consistent with those for a random vector solenoidal field. Some final comments are presented in Section 5.

## 2 Lattice Boltzmann algorithm for MHD

It is convenient to write the MHD equations in conservation form

$$\frac{\partial(\rho u_i)}{\partial t} + \frac{\partial}{\partial x_j} \left[ \rho u_i u_j - B_i B_j + \left( p + \frac{1}{2} \mathbf{B}^2 \right) \delta_{ij} \right] = \nu \nabla^2 u_i, \quad (2.1)$$

$$\frac{\partial B_i}{\partial t} + \frac{\partial}{\partial x_j} [u_i B_j - u_j B_i] = \eta \nabla^2 B_i, \quad (2.2)$$

where  $\rho$  is the density,  $\mathbf{u}$  the fluid velocity,  $\mathbf{B}$  the magnetic field,  $\nu$  the viscosity and  $\eta$  the resistivity. The summation convention over repeated subscripts is employed. The resistive MHD equations are closed by continuity and an isothermal equation of state:

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \mathbf{u}) = 0, \quad \nabla \cdot \mathbf{B} = 0, \quad p = c_s^2 \rho = \frac{1}{3} \rho. \quad (2.3)$$

### 2.1 LB representation for Navier-Stokes turbulence

We now summarize the LB representation for Navier-Stokes turbulence with  $\mathbf{B} = 0$  in Eqs. (2.1)-(2.2). It is well known that to recover the 3D Navier-Stokes equations from a discrete kinetic representation [3–11] one should use a lattice geometry on the unit cube that is at least D3Q15, D3Q19 or D3Q27, where these lattices are so designated by their lattice vectors  $\mathbf{e}_\alpha$ ,  $\alpha = 1, \dots, Q$

$$\begin{aligned} 15\text{-bit: D3Q15: speeds } 0, 1, \sqrt{3}: & (0,0,0), (0,0,\pm 1), (\pm 1, \pm 1, \pm 1) \\ 19\text{-bit: D3Q19: speeds } 0, 1, \sqrt{2}: & (0,0,0), (0,0,\pm 1), (0, \pm 1, \pm 1) \\ 27\text{-bit: D3Q27: speeds } 0, 1, \sqrt{2}, \sqrt{3}: & (0,0,0), (0,0,\pm 1), (0, \pm 1, \pm 1), (\pm 1, \pm 1, \pm 1) \end{aligned} \quad (2.4)$$

with appropriate permutations. A plot of the D3Q19 lattice is shown in Fig. 1.

The underlying lattice geometry is so chosen that the lattice vectors have the following isotropy properties to at least the 4<sup>th</sup> order moments:

$$\begin{aligned} \sum_{\alpha=1}^Q e_{\alpha i} &= 0, \quad \sum_{\alpha=1}^Q e_{\alpha i} e_{\alpha j} = \alpha_Q \delta_{ij}, \quad \sum_{\alpha=1}^Q e_{\alpha i} e_{\alpha j} e_{\alpha k} = 0, \\ \sum_{\alpha=1}^Q e_{\alpha i} e_{\alpha j} e_{\alpha k} e_{\alpha l} &= b_Q (\delta_{ij} \delta_{kl} + \delta_{ik} \delta_{jl} + \delta_{il} \delta_{jk}), \end{aligned} \quad (2.5)$$

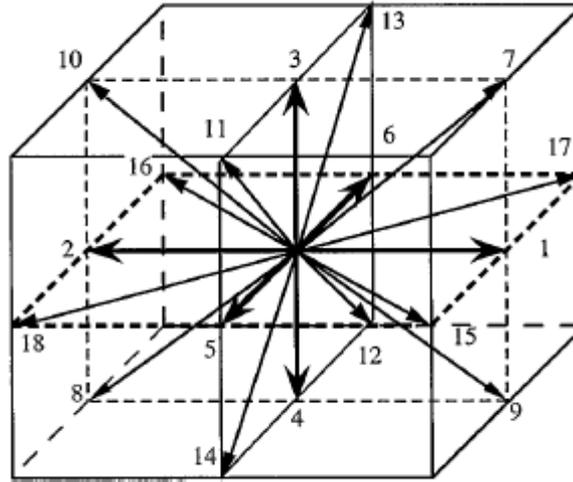


Figure 1: The D3Q19 lattice on the unit cube with lattice vectors  $\mathbf{e}_\alpha$ ,  $\alpha = 1, \dots, Q = 15$ . There is one "speed 0" (rest particle), six "speed 1" along the axes ("1" - "6") and twelve "speed  $\sqrt{2}$ " along the planar diagonals ("7" - "19").

where  $a_Q$  and  $b_Q$  are lattice-dependent constants. All odd moments of the lattice vectors are zero (by symmetry).

The discretized LB kinetic equation for the Navier-Stokes equation, written in LB units of  $|\Delta \mathbf{x}| = 1 = \Delta t$ , is

$$f_\alpha(\mathbf{x} + \mathbf{e}_\alpha, t + 1) - f_\alpha(\mathbf{x}, t) = -\frac{1}{\tau_u} [f_\alpha(\mathbf{x}, t) - f_\alpha^{eq}(\rho[\mathbf{x}, t], \mathbf{u}[\mathbf{x}, t])], \quad \alpha = 1, \dots, Q. \quad (2.6)$$

The standard (discrete) moments connect the distribution function to the macroscopic variables:

$$\sum_{\alpha=1}^Q f_\alpha = \rho, \quad \sum_{\alpha=1}^Q f_\alpha \mathbf{e}_\alpha = \rho \mathbf{u}, \quad \sum_{\alpha=1}^Q f_\alpha e_{\alpha i} e_{\alpha j} = \frac{\rho}{3} \delta_{ij} + \rho u_i u_j. \quad (2.7)$$

The relaxation rate  $\tau_u$ , through the Chapman-Enskog long-wavelength and long-time asymptotics [3, 4], determines the viscosity and hence the Reynolds number

$$\nu = \frac{1}{6}(2\tau_u - 1), \quad (2.8)$$

$$\text{Re} = \frac{U_0 L}{\nu}, \quad (2.9)$$

for some mean velocity  $U_0$  and mean length scale  $L$ . The discrete (in the kinetic velocity space) lattice symmetry must be such that to leading order it does not lead to any unphysical (i.e., non-Galilean invariant) effects at the  $\mathcal{O}(3)$  continuous symmetry of the macroscopic nonlinear system one is modeling. It [3–11] can be shown that an appropri-

ate polynomial representation for the relaxation distribution function is

$$f_{\alpha}^{eq}[\rho, \mathbf{u}] = \rho w_{\alpha} \left[ 1 + 3\mathbf{e}_{\alpha} \cdot \mathbf{u} + \frac{9}{2}(\mathbf{e}_{\alpha} \cdot \mathbf{u})^2 - \frac{3}{2}\mathbf{u} \cdot \mathbf{u} + \frac{9}{2}(\mathbf{e}_{\alpha} \cdot \mathbf{u}) \{(\mathbf{e}_{\alpha} \cdot \mathbf{u})^2 - \mathbf{u} \cdot \mathbf{u}\} \right] + \mathcal{O}(u^4). \quad (2.10)$$

This form is invariant to the particular lattice geometry — whether D3Q15, D3Q19 or D3Q27. The weight factors  $w_{\alpha}$  are normalized

$$\sum_{\alpha=1}^Q w_{\alpha} = 1, \quad (2.11)$$

but are lattice dependent — see Table 1. The weight factor  $w_{\alpha}$  in Eq. (2.11) have subscripts referring to the lattice directions, but in Table 1 we use a convention where the weight factor is written as  $w_{E(\alpha)}$ , with its subscript equal to the respective particle energy, where  $E(\alpha) = 0, 1, 2, 3$  for  $\alpha = 1, 2, \dots, Q$  according to the lattice stencil in Fig. 1.

Table 1: The dependence of the weight factor  $w_{E(\alpha)}$  on the lattice symmetry. The subscript of the weight factor is the value of particle kinetic energy (velocity squared with  $m=1$ ) which depends on  $\alpha$ .

Speed (# velocities)	D3Q15	D3Q19	D3Q27
0 (1 bit)	$w_0 = \frac{2}{9}$	$w_0 = \frac{1}{9}$	$w_0 = \frac{8}{27}$
1 (6 bits)	$w_1 = \frac{1}{9}$	$w_1 = \frac{1}{18}$	$w_1 = \frac{2}{27}$
$\sqrt{2}$ (12 bits)	-	$w_2 = \frac{1}{36}$	$w_2 = \frac{1}{54}$
$\sqrt{3}$ (8 bits)	$w_3 = \frac{1}{72}$	-	$w_3 = \frac{1}{216}$

The beauty of LB is in its simplicity: the algorithm to solve Eqs. (2.6) and (2.11) is just kinetic streaming the distribution values to the appropriate nearby lattice followed by local node collisional relaxation. The macroscopic nonlinearities in the relaxation distribution function are purely local but at the macroscopic level will become (under Chapman-Enskog expansions) the convective nonlinear derivatives of the Navier-Stokes equation. Moreover, in the spatial domain decomposition, the only MPI communication between neighboring processors is in the streaming of information from boundary nodes.

However, the LB algorithm while an explicit second accurate scheme, is subject to numerical instabilities as  $\tau_u \rightarrow 0.5_+$ . These instabilities arise because there are no constraints imposed such that  $f_{\alpha} \geq 0$ . This defect has been remedied by the introduction of entropic lattice Boltzmann schemes [5–11] for Navier-Stokes flows.

## 2.2 ELB representation of Navier-Stokes turbulence

To ensure positive definiteness of the distribution functions  $f_{\alpha} \geq 0$  one introduces a non-decreasing Lyapunov function

$$H = \sum_{\alpha=1}^Q h_{\alpha}(f_{\alpha}) \quad (2.12)$$

for some to-be-determined convex functions  $h_\alpha$ , with  $h'_\alpha \geq 0$ . On extremizing the Lyapunov function subject to the local collisional invariants of mass and momentum, one can determine the associated Lagrange multipliers analytically for low Mach number flows for the D3Q27-lattice (but not for the D3Q15 or D3Q19 lattices [11]). In particular, one can determine the unique form of the Lyapunov function to be

$$H[f_\alpha] = \sum_{\alpha=1}^Q f_\alpha \ln \left( \frac{f_\alpha}{w_\alpha} \right), \quad (2.13)$$

where the weights  $w_\alpha$  are exactly those in Table 1, and with the relaxation distribution function  $f_\alpha^{eq}$

$$f_\alpha^{eq}[\rho, \mathbf{u}] = \rho w_\alpha \prod_{i=1}^3 \left( 2 - \sqrt{1 + 3u_i^2} \right) \left( \frac{2u_i + \sqrt{1 + 3u_i^2}}{1 - u_i} \right)^{e_{\alpha,i}}, \quad \alpha = 1, \dots, 27. \quad (2.14)$$

A low Mach number expansion of Eq. (2.14) recovers the standard polynomial form Eq. (2.10) for the D3Q27 lattice.

The generalized LB algorithm now becomes a two-parameter BGK scheme

$$f_\alpha(\mathbf{x} + \mathbf{e}_\alpha, t+1) - f_\alpha(\mathbf{x}, t) = -\frac{\gamma(\mathbf{x}, t)}{2\tau_u} [f_\alpha(\mathbf{x}, t) - f_\alpha^{eq}(\rho[\mathbf{x}, t], \mathbf{u}[\mathbf{x}, t])], \quad \alpha = 1, \dots, Q, \quad (2.15)$$

where the function  $\gamma(\mathbf{x}, t)$  is the nontrivial root of the Lyapunov equation

$$H[\mathbf{f}] = H[\mathbf{f} - \gamma(\mathbf{f} - \mathbf{f}^{eq})]. \quad (2.16)$$

From Eq. (2.15), with  $2\tau_u = 1$ , one sees that Eq. (2.16) simply states that the post-collision distributions lie on the same constant entropy surface as the pre-collision distributions. Thus, the enforcement of Eq. (2.15), with  $2\tau_u = 1$ , yields a reversible detailed-balance algorithm which must be unconditionally stable. Nevertheless, at the macroscopic level the system recovered by Chapman-Enskog expansions is irreversible: the Navier-Stokes equation with generalized viscosity

$$\nu_{eff}(\mathbf{x}, t) = \frac{1}{6} \left[ \frac{4\tau_u}{\gamma(\mathbf{x}, t)} - 1 \right]. \quad (2.17)$$

Moreover, we [11] have shown that the ELB-algorithm can be readily extended to both the D3Q15 and D3Q19 models, using the polynomial equilibria of Eq. (2.10) and the appropriate weights given in Table 1. Notice also that even if the ELB algorithm was run in detailed-balance form [i.e.,  $2\tau_u = 1$ ], there is still some effective viscosity at the macroscopic level  $\nu_{eff}(\mathbf{x}, t) = 2 - \gamma(\mathbf{x}, t) / 6\gamma(\mathbf{x}, t)$  even though the bare molecular viscosity, Eq. (2.8), is zero:  $\nu = 0$ .

### 2.3 LB representation for MHD turbulence

Dellar, in a seminal paper [2], extended LB to MHD by introducing a vector distribution function for the magnetic field  $\mathbf{B}(\mathbf{x}, t)$ . While the mean velocity is generated by the first moment of the scalar distribution  $f_\alpha$ , the magnetic field is generated by the zeroth moment of the vector distribution function  $\mathbf{g}_\beta$ :

$$\sum_{\alpha=1}^Q f_\alpha \mathbf{e}_\alpha = \rho \mathbf{u}; \quad \sum_{\beta=1}^{Q'} \mathbf{g}_\beta = \mathbf{B}. \quad (2.18)$$

If the MHD equations are written in conservative form (in the isothermal limit)

$$\frac{\partial}{\partial t} (\rho u_i) + \frac{\partial}{\partial x_j} \left[ \left( \frac{\rho}{3} + \frac{1}{2} \mathbf{B}^2 \right) \delta_{ij} + \rho u_i u_j - B_i B_j \right] = \nu \nabla^2 u_i, \quad (2.19)$$

$$\frac{\partial B_i}{\partial t} + \frac{\partial}{\partial x_j} [u_i B_j - u_j B_i] = \eta \nabla^2 B_i, \quad (2.20)$$

one immediately notes that the momentum stress tensor  $\Pi_{ij}$  in Eq. (2.19) is symmetric in  $i \leftrightarrow j$  interchange

$$\Pi_{ij} = \left( \frac{\rho}{3} + \frac{1}{2} \mathbf{B}^2 \right) \delta_{ij} + \rho u_i u_j - B_i B_j, \quad (2.21)$$

while the magnetic stress tensor  $\Lambda_{ij}$  in Eq. (2.20) is antisymmetric

$$\Lambda_{ij} = u_i B_j - u_j B_i. \quad (2.22)$$

These symmetries are readily preserved if the momentum stress tensor is generated by the second (symmetric) moment of  $f_\alpha$  while the magnetic stress tensor is generated by the first moment of  $\mathbf{g}_\beta$ :

$$\Pi_{ij} = \sum_{\alpha=1}^Q f_\alpha e_{\alpha i} e_{\alpha j} = \Pi_{ji}, \quad \Lambda_{ij} = \sum_{\beta=1}^{Q'} g_{\beta i} e_{\beta j} = -\Lambda_{ji}. \quad (2.23)$$

We should note that the underlying lattice determining the magnetic field could have lower symmetry than that for determining the mean velocity field since the moment closure level for the magnetic field is at first order while that for the mean velocity field is at second order. This is why we write the upper bound on the magnetic distribution function moments as  $Q'$  rather than  $Q$ .

It can be shown that the following choice of relaxation distribution functions leads to the MHD equations (2.1)-(2.3) in the Chapman-Enskog limit

$$f_\alpha^{eq}[\rho, \mathbf{u}, \mathbf{B}] = \rho w_\alpha \left[ 1 + 3 \mathbf{e}_\alpha \cdot \mathbf{u} + \frac{9}{2} (\mathbf{e}_\alpha \cdot \mathbf{u})^2 - \frac{3}{2} \mathbf{u}^2 \right] + \frac{9}{2} w_\alpha \left[ \frac{1}{2} \mathbf{B}^2 \mathbf{e}_\alpha^2 - (\mathbf{e}_\alpha \cdot \mathbf{B})^2 - \frac{1}{6} \mathbf{B}^2 \right], \quad (2.24)$$

$$g_{\beta i}^{eq}[\mathbf{u}, \mathbf{B}] = w_\beta [B_i + e_{\beta j} (u_j B_i - u_i B_j)],$$

where  $\alpha, \beta = 15, 19$  or  $27$ . To keep the memory requirements in the numerical simulations to manageable levels (since each processor/core has typically 2 GB of memory) we will work with a  $Q = 27$ -bit scalar distribution  $f_\alpha$  and a  $Q' = 15$ -bit *vector* distribution  $\mathbf{g}_\beta$ .

### 3 Parallelization of LB algorithms

The basic structure of LB algorithms consists of two conceptual steps, shown below for the MHD case (in the case of ELB algorithm, of course, the vector magnetic distribution function is absent):

- (a) local collisional calculation at  $(\mathbf{x}, t)$ , with  $\mathbf{u}$  and  $\mathbf{B}$  also being determined locally by simple moments:

$$\begin{aligned} f_\alpha(\mathbf{x}, t) - \frac{1}{\tau_u} [f_\alpha(\mathbf{x}, t) - f_\alpha^{eq}(\rho[\mathbf{x}, t], \mathbf{u}[\mathbf{x}, t], \mathbf{B}[\mathbf{x}, t])] &\rightarrow f'_\alpha(\mathbf{x}, t), \quad \alpha = 1, \dots, 27, \\ \mathbf{g}_\beta(\mathbf{x}, t) - \frac{1}{\tau_B} [\mathbf{g}_\beta(\mathbf{x}, t) - \mathbf{g}_\beta^{eq}(\mathbf{u}[\mathbf{x}, t], \mathbf{B}[\mathbf{x}, t])] &\rightarrow \mathbf{g}'_\beta(\mathbf{x}, t), \quad \beta = 1, \dots, 15; \end{aligned} \quad (3.1)$$

- (b) streaming these post-collision distributions to the nearby spatial nodes to complete the update to time  $t + 1$ :

$$\begin{aligned} f'_\alpha(\mathbf{x}, t) &\rightarrow f_\alpha(\mathbf{x} + \mathbf{e}_\alpha, t + 1), \quad \alpha = 1, \dots, 27, \\ \mathbf{g}'_\beta(\mathbf{x}, t) &\rightarrow \mathbf{g}_\beta(\mathbf{x} + \mathbf{e}_\beta, t + 1), \quad \beta = 1, \dots, 15. \end{aligned} \quad (3.2)$$

It is pointed out that (a) is computationally intensive, but requires only data local to the grid point; (b) is a set of shift operations, moving data from grid point to grid point according to the lattice vector.

However, a key optimization [13] is that the two stages (a) and (b) can be partially combined — either the newly calculated post-collision distribution function is streamed immediately to its new position as soon as it is calculated, or data can be gathered from adjacent cells to calculate the updated value for the current cell. The memory access pattern for the collision phase becomes more complex, but the amount of data transferred at each time step is reduced. Typically, this is 20-30% faster than implementing the two steps separately. All results reported here are based on code implementations that make use of this optimization.

The optimal layout for the distribution functions for most computer architectures [13] has the first dimensions to be the Cartesian grid points followed by the index representing the streaming vector:  $f(x, y, z, 27)$  and  $\mathbf{g}(x, y, z, 15, 3)$ . Fortran array syntax is assumed with  $x$  varying fastest when stepping contiguously through memory. The basic structure of both LB applications consists of three nested loops over spatial grid points (typically 100 sec iterations per loop) with inner loops over velocity streaming vectors and, in the case of MHD, magnetic field streaming vectors (typically 10 sec iterations). Within these

innermost loops the various macroscopic quantities and their updated values are calculated via various algebraic expressions.

For ELB, a non-linear equation must be solved for each grid-point and at each time-step so that the collision process satisfies the constraints described in Section 2.2. The equation is solved via Newton-Raphson iteration (5 iterations are usually enough to converge to within  $10^{-8}$ ), and as this equation involves taking the logarithm of each component of the distribution function at each iteration, the whole algorithm become heavily constrained by the performance of the log function.

For the MHD case, on the ES, the innermost loops were unrolled via compiler directives and the (now) innermost grid point loop was vectorized. This proved a very effective strategy, boosting performance from 330 MFlop/s, where the phase space loops were vectorized, to 3.97 GFlop/s. For the superscalar architectures, Power5, BG/L, and XT3, we did not explicitly tune further for any architecture.

In Table 2 we present our most optimized performance on the ES, where the somewhat unusual domain decomposition was chosen to keep the vector length (length of  $x$  dimension) long.

Table 2: Performance of the 3D-LBMHD code on the Earth Simulator (a 5120 CPU vector machine with CPU peak of 8 GFlops/s). Our 4800 CPU run achieved 26.25 TFlops/s, 67% of peak on the ES, and outputted 250 GB of data.

GRID	#CPU	DOMAIN DECOMP	%MPI	AVG. MSG SIZE (MB)	VECTOR LENGTH	GFLOPS/S/ CPU
$512^3$	256	$2 \times 8 \times 16$	7.7	2.1	254.2	5.43
$512^3$	512	$2 \times 16 \times 16$	9.1	1.1	253.2	5.19
$1024^3$	1024	$4 \times 16 \times 16$	5.1	2.3	254.5	5.44
$1024^3$	2048	$4 \times 16 \times 32$	8.6	2.1	254.5	5.36
$1024^3$	4096	$4 \times 32 \times 32$		1.1	253.3	5.16
$1440^3$	4800	$2 \times 40 \times 60$			239.8	5.47

For ELB, in the case of the Earth Simulator, the compiler was able to vectorize all loops containing the log functions in determining the entropy surface. The routine containing the nonlinear equation solver was rewritten to operate on an array of grid points, rather than a single point, allowing vectorization of this recursive operation. After this optimization, high performance was achieved.

For the superscalar systems, using the rewritten vector version of non-linear equation solving routine proved to be much faster than the original approach. Presumably this is due to a reduction of routine-call overhead and better use of the functional units. Depending on the architecture, a speedup of 20-30% is achieved on switching to the new routine. Another important optimization was to use optimized library routines to compute a vector of logarithm values per invocation. Each architecture offers an optimized math function library: MASS for IBM Power5 and BG/L; and ACML for AMD Opteron of the XT3. A 15-30% speedup over the "non-vector" log function is achieved.

For parallel implementation each array is partitioned onto a 3D Cartesian processor grid, and MPI is used for communication. Ghost cells are used to hold copies of the planes of data from neighboring processors. There are two obvious methods of exchanging data with neighbors: communicate with all 26 neighboring cells, or use the shift method [14] where we first exchange only in one direction and so partially populate the ghost cells. The next exchange includes this data, further populating the ghost cells, until all the data has been exchanged. The disadvantage of the shift algorithm is that it requires synchronization after each pair of exchanges, i.e. there are only four messages in flight at any time. However, it has the distinct advantage of increasing the average message size, and on most parallel architectures the message latency is relatively high compared with the bandwidth.

Because different lattice vectors contribute to different spatial directions, the data to be exchanged are not contiguous in the  $f$  or  $g$  arrays. For example, in the 27-bit model, 12 of the 26 lattice vectors have a component in the  $+x$ -direction and must be exchanged in this direction, but are not contiguous in the velocity distribution function array  $f$ . The data is packed into a single buffer, resulting in 6 message exchanges per time step, using *mpi\_isend/mpi\_irecv* pairs.

In Fig. 2 we present the scaling of both the MHD and the ELB codes on various architectures (both vector and scalar) For each set of results, we use the maximum grid size possible within the memory constraints of each architecture, basically a weak scaling experiment. Not only is the scaling of the sustained TFlops/s with concurrency excellent — no saturation is seen, even up to 32,768 cores on BG/L, but the flop-rate itself is impressive. The 11% of peak on BlueGene is calculated using the peak of the SIMD double hummer FPU, even though the SIMD unit can only be utilized under very special conditions.

The LB-MHD simulations reported here were run on an SGI Altix 4200 at the Air Force Research Laboratory with excellent scaling with cores, as seen in Fig. 3. We plot the total CPU time for two different grids as we scale up with cores. For the  $1024^3$ -grid we used 1024 to 8192 cores and saw the wallclock time halved as we doubled the number of cores, resulting in 'perfect' scaling with processors. On the other hand, for the  $1800^3$ -grid, we used both 4500 and the maximum 9000 cores and saw superlinear scaling: the wallclock time was significant more than halved by doubling the number of cores. This is attributable to more of the simulation data can be held in cache memory.

## 4 Lattice Boltzmann simulations

### 4.1 ELB for Navier-Stokes turbulence

The ELB algorithm [3–11, 15] is unconditionally stable and we briefly present here some simulations results to indicate its potential in fluid turbulence and to thus emphasize the importance of developing an analogous entropic algorithm for MHD. For Navier-Stokes flows, we [15] consider free decaying turbulence from an initial Kida [16] incompressible

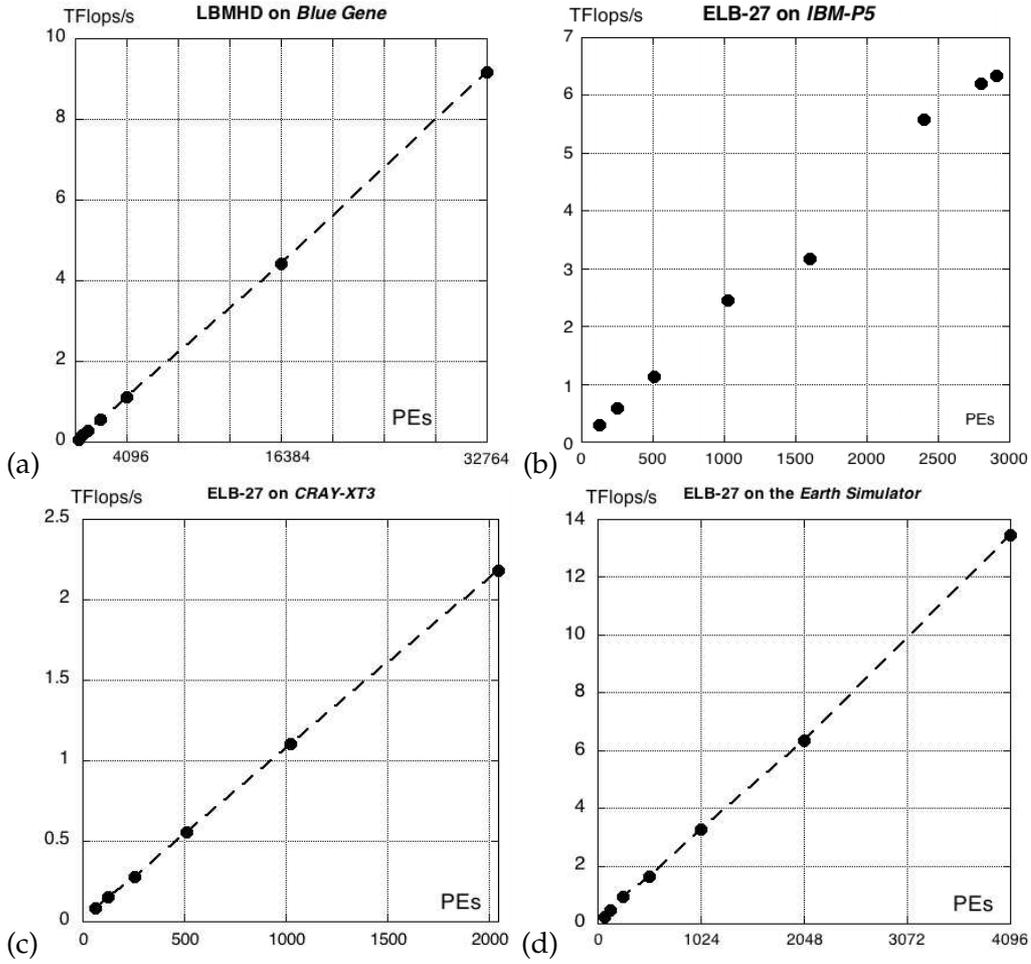


Figure 2: (a):BlueGene: 32,768 PEs, 9.1 TFlops/s, 11% peak; (b):IBM-P5: 2912 PEs, 6.3 TFlops/s, 30% peak; (c):CRAY-XT3:2048 PEs, 2.2 TFlops/s, 23% peak; (d) ES:4084 PEs, 13.5 TFlops/s, 41% peak. The performance of Lattice Boltzmann codes on various architectures (a) MHD code on BlueGene, (b)-(d) entropic Navier-Stokes on the IBM-P5, CRAY-XT3 and Earth Simulator. Not only is the scaling of the sustained TFlops/s with PEs is excellent — no saturation is seen, even up to 32,768 cores on BlueGene, but the flop-rate itself is impressive. The 11% on BlueGene is calculated using their 2 FPU units, even though the second one can only be utilized under very special conditions. It is thus very common to quote timings as if there were only 1 FPU — in which case our performance on BlueGene would be 22% of peak.

highly symmetric velocity profile on a  $2\pi^3$ -grid:

$$\begin{aligned}
 u_x(x,y,z,t=0) &= U_0 [\cos 3y \cos z - \cos y \cos 3z] \sin x, \\
 u_y(x,y,z,t=0) &= U_0 [\cos 3z \cos x - \cos z \cos 3x] \sin y, \\
 u_z(x,y,z,t=0) &= U_0 [\cos 3x \cos y - \cos x \cos 3y] \sin z.
 \end{aligned} \tag{4.1}$$

Initially this profile has zero local helicity everywhere:  $\mathbf{u}(\mathbf{x},0) \cdot \boldsymbol{\omega}(\mathbf{x},0) = 0$ , where  $\boldsymbol{\omega}$  is the vorticity  $\boldsymbol{\omega} = \nabla \times \mathbf{u}$ . On a spatial grid of  $1600 \times 1600 \times 1600$ , and at Reynolds number

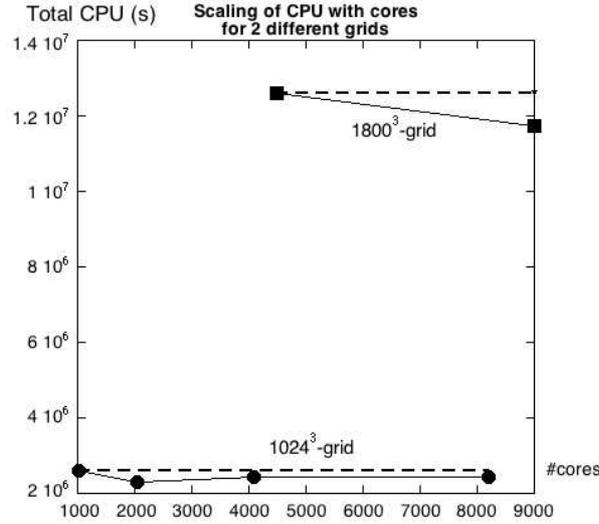


Figure 3: The scaling of the LB-MHD code on 1024<sup>3</sup>-grid and 1800<sup>3</sup>-grid on the SGI Altix. The dashed curve represents 'perfect' scaling: doubling the number of cores exactly halves the wallclock time. Plotted on the  $y$ -axis is the total CPU-time (= wallclock time  $\times$  #cores) which would thus remain constant for 'perfect' scaling. The 1024<sup>3</sup>-grid showed such scaling as we increased the number of cores from 1024 to 8192. The 1800<sup>3</sup>-grid run showed even better scaling due to excellent use of cache memory, resulting in superlinear scaling: the total CPU-time significantly decreased as the cores were scaled from 4500 to the maximum 9000 available.

Re  $\approx$  25000 we plot in Fig. 4 the time evolution of the 1D energy spectra

$$E_{long}(k_x, t) = \sum_{k_y, k_z} |u_x(k_x, k_y, k_z, t)|^2, \quad E_{trans}(k_x, t) = \sum_{k_y, k_z} |u_y(k_x, k_y, k_z, t)|^2 \quad (4.2)$$

from the  $\delta$ -function energy spectra initial conditions

$$E_{long}(k_x, t=0) = E_{long} \delta(k_x - 2), \quad E_{trans}(k_x, t=0) = E_{trans} [\delta(k_x - 2) + \delta(k_x - 4)] \quad (4.3)$$

towards a Kolmogorov inertial  $k_x^{-5/3}$  spectrum.

The simulation is basically fully resolved at all the scales excited by the turbulence since there is only a very slight upturn in the longitudinal energy spectrum at the very end of the  $k_x$ -spectrum. There is an even smaller upturn in the transverse energy spectrum, Fig. 4b. The time-dependent microscale Reynolds number for the simulation is shown in Fig. 5, with

$$\text{Re}_\lambda(t) = \left(\frac{20}{3}\right)^{1/2} \frac{\int_0^\infty dk(k, t)}{\nu (\int_0^\infty dk k^2 E(k, t))^{1/2}} = \left(\frac{20}{3}\right)^{1/2} \frac{E(t)}{\nu_\Omega^{1/2}(t)}, \quad (4.4)$$

where  $E(t)$  is the kinetic energy,  $\Omega(t)$  the entropy and  $\nu$  the molecular viscosity.

In Fig. 6 we look in closer detail at the 1D transverse energy spectrum at  $t=28K$  and  $t=38K$ . It is seen that the energy spectrum is better fitted by  $k_x^{-5/3-0.1}$  than the Kolmogorov energy spectrum. This shift from the  $k_x^{-5/3}$  is an indicator of intermittency.

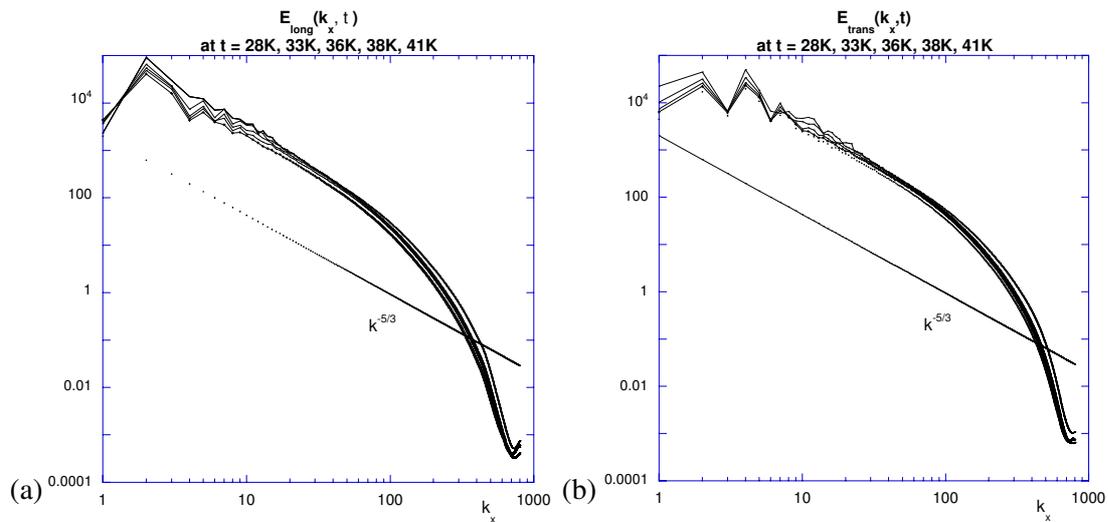


Figure 4: (a) The 1D longitudinal and (b) the 1D transverse energy spectra at times  $t = 28K, 33K, 36K, 38K, 41K$  and  $54K$  as a function of wavenumber  $k_x$ . The dashed green line is the Kolmogorov  $k^{-5/3}$  inertial energy [15].

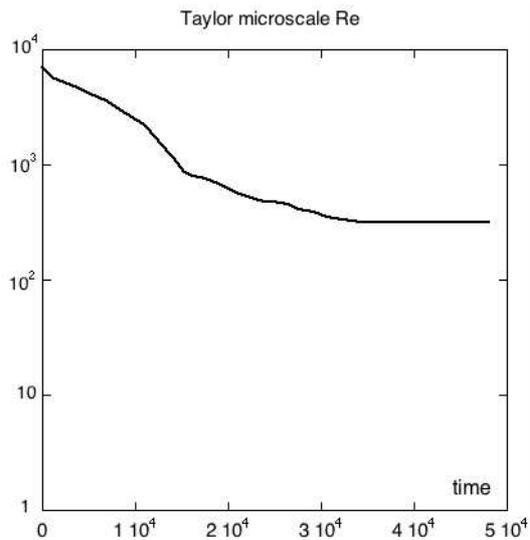


Figure 5: The evolution of the time-dependent Taylor microscale Reynolds number for the free-decaying turbulence simulation.

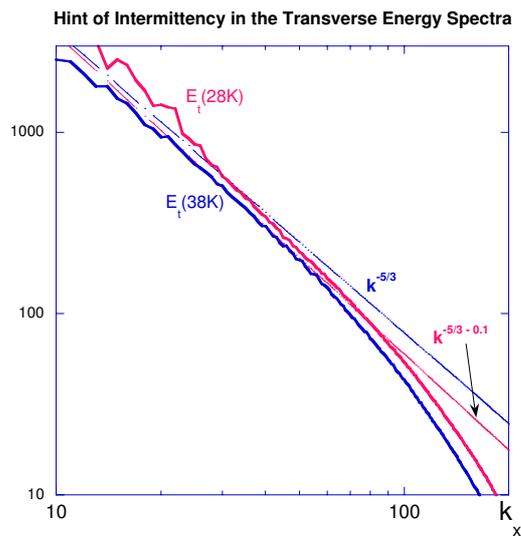
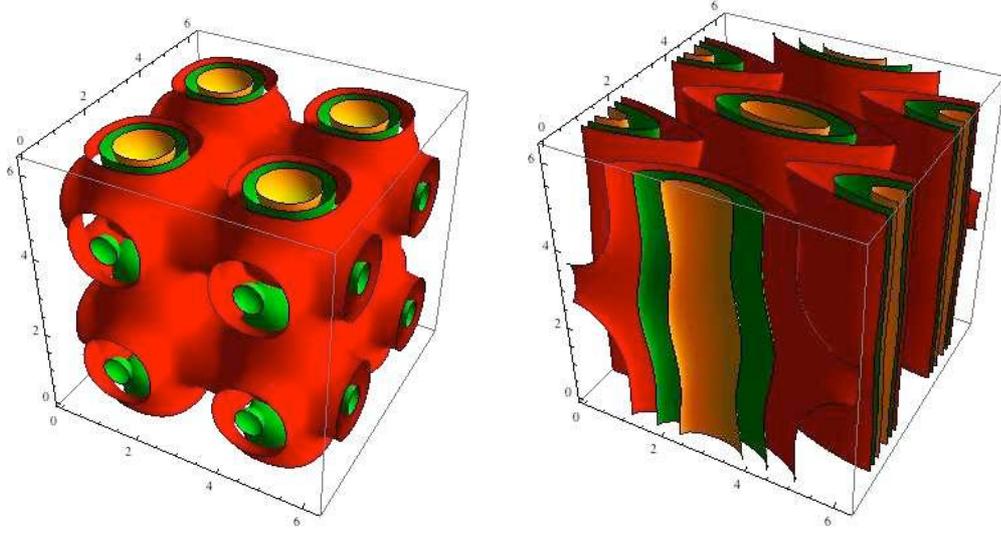


Figure 6: The 1D transverse energy spectrum at times  $t = 28K$  and  $38K$  which exhibits a  $k_x^{-5/3-0.1}$  scaling. This deviation from the Kolmogorov is indicative of intermittency [12].

As first noted by Kaneda et al. [12] on a pseudo-spectral  $4800^3$ -grid with judicious use of single/double precision. It is interesting to note that we have been able to discern this sign of intermittency with an entropic lattice Boltzmann simulation on a  $1600^3$ -grid.

Figure 7: (a) Isosurfaces of  $|\text{Vorticity}|$ ; (b) Isosurfaces of  $|\text{Current}|$ .

## 4.2 LB for MHD turbulence

We now consider the lattice Boltzmann simulation for MHD using Eq. (2.24) for the velocity relaxation distribution functions with  $f_{\alpha}^{eq}, \alpha = 1, \dots, Q = 27$  and for the magnetic relaxation distribution function  $g_{\beta}^{eq}, \beta = 1, \dots, Q' = 15$ . The initial profiles were a Taylor-Green velocity profile in an Orszag-Tang magnetic field:

$$\begin{aligned} \mathbf{u}(\mathbf{x}, t=0) &= U_0(\sin x \cos y \cos z, -\cos x \sin y \cos z, 0), \\ \mathbf{B}(\mathbf{x}, t=0) &= B_0(-2\sin 2y + \sin z, 2\sin x + \sin z, \sin x + \sin y). \end{aligned} \quad (4.5)$$

The corresponding isosurfaces of vorticity and current shown in Fig. 7.

With these profiles, there is no initial magnetic helicity or cross helicity

$$0 = \int d^3x \mathbf{A}(\mathbf{x}, 0) \cdot \mathbf{B}(\mathbf{x}, 0), \quad 0 = \int d^3x \mathbf{u}(\mathbf{x}, 0) \cdot \mathbf{B}(\mathbf{x}, 0), \quad (4.6)$$

where  $\mathbf{A}$  is the vector potential.

An extremely important property of the LB algorithm for MHD [1,2,17,18] is that from the Chapman-Enskog expansions one can show that the trace of the first order magnetic stress tensor is proportional the divergence of the magnetic field — and hence this must be zero since the magnetic stress tensor is antisymmetric:

$$0 = \text{Tr} \Lambda^{(1)} = \sum_{\alpha, i} e_{\alpha i} [g_{\alpha i} - g_{\alpha i}^{eq}] = -\frac{\tau_B}{3} \nabla \cdot \mathbf{B}. \quad (4.7)$$

We have verified this result directly from our LB simulations by explicitly calculating the trace of the magnetic stress tensor, Fig. 8, with  $\text{Tr} \Lambda^{(1)} = 0$  to machine accuracy.

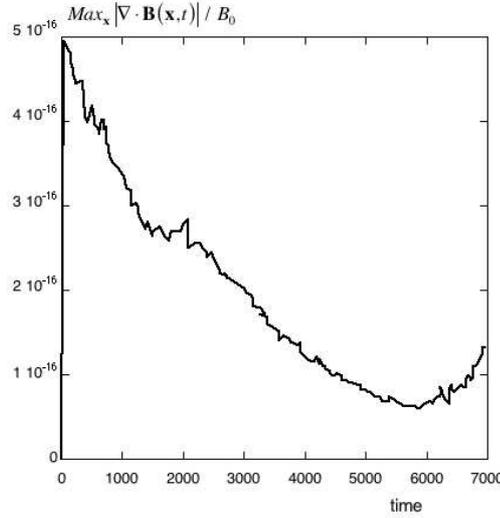


Figure 8: The time evolution of  $Tr\Lambda^{(1)}(t) = \sum_{\alpha,i} e_{\alpha i} (g_{\alpha i} - g_{\alpha i}^{eq})$  in the LB simulation, showing it is 0 to machine accuracy. Chapman-Enskog asymptotics yields  $Tr\Lambda^{(1)} = -\tau_B \nabla \cdot \mathbf{B}/3$ .

We present results from two large simulation runs on a spatial lattice of  $1800 \times 1800 \times 1800$  using all the 9000 cores available on the SGI Altix. The first run ran for 60K time steps with

$$\text{Case A: } Re = \frac{U_0 L}{\nu} = 1000, \quad Rm = \frac{B_0 L}{\eta} = 350, \quad Pr = \frac{\nu}{\eta} = 0.3, \quad (4.8)$$

while the second run ran to 30K time steps at a higher Prandtl and magnetic Reynolds number

$$\text{Case B: } Re = \frac{U_0 L}{\nu} = 350, \quad Rm = \frac{B_0 L}{\eta} = 1050, \quad Pr = \frac{\nu}{\eta} = 3.0. \quad (4.9)$$

In Fig. 9 we plot the time development of the normalized energies, enstrophies and palinstrophy

$$\begin{aligned} E_{kin}(t) &= \int d^3x \mathbf{u}^2(\mathbf{x}, t), & E_{mag}(t) &= \int d^3x \mathbf{B}^2(\mathbf{x}, t), \\ \text{Kinetic Enstrophy } \Omega(t) &= \int d^3x |\nabla \times \mathbf{u}(\mathbf{x}, t)|^2 = \langle \omega^2(\mathbf{x}, t) \rangle, \\ \text{Magnetic Enstrophy } \Omega_M(t) &= \int d^3x |\nabla \times \mathbf{B}(\mathbf{x}, t)|^2 = \langle \mathbf{J}^2(\mathbf{x}, t) \rangle, \\ \text{Palinstrophy } (t) &= \int d^3x |\nabla \times \omega(\mathbf{x}, t)|^2 = \langle |\nabla \times \omega(\mathbf{x}, t)|^2 \rangle, \end{aligned} \quad (4.10)$$

which are just higher order  $k$ -moments of the energy spectra.

In Navier-Stokes turbulence [11, 12, 15], the kinetic enstrophy increases sharply at early times due to inviscid vortex stretching — i.e., the kinetic enstrophy increase is independent of the transport coefficient. The kinetic energy during this period is very slowly

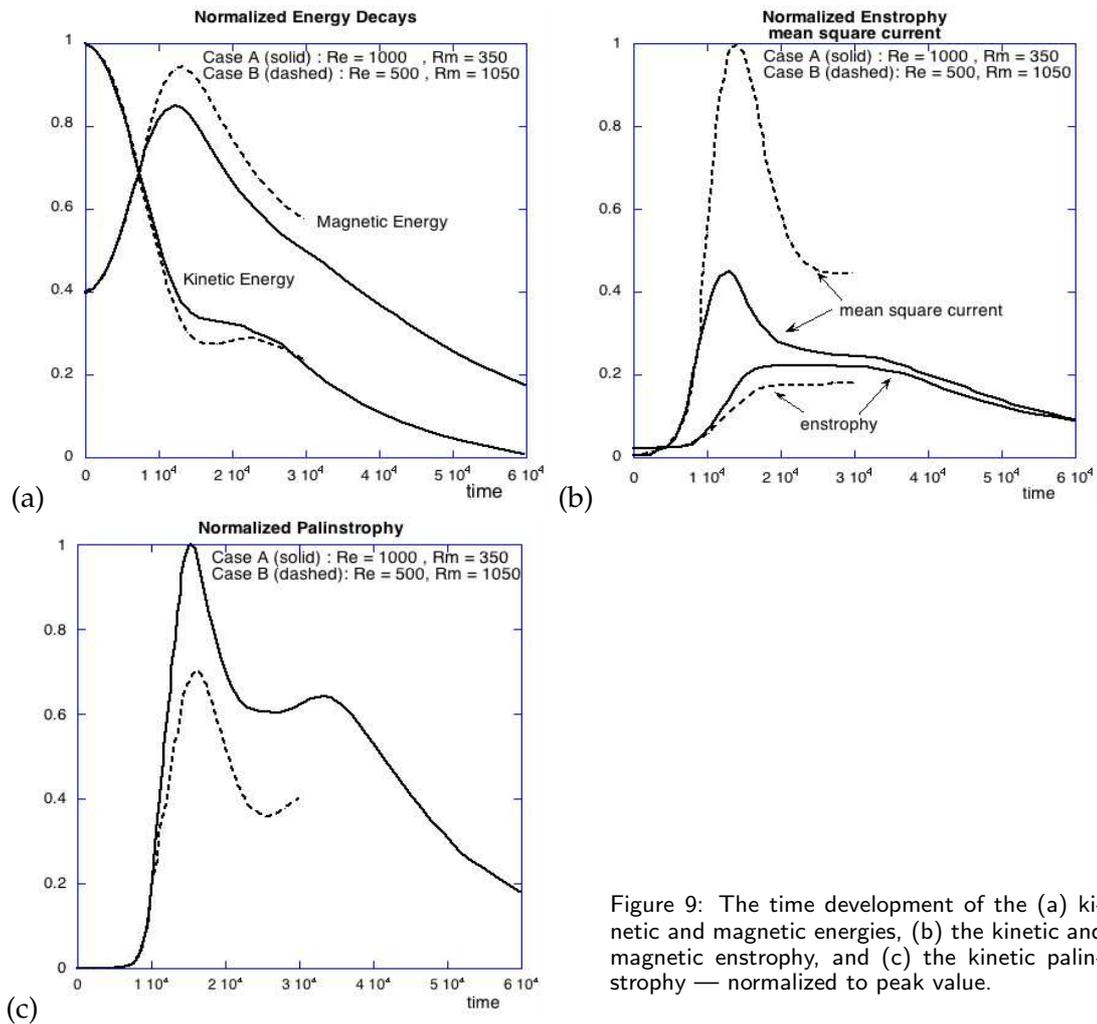


Figure 9: The time development of the (a) kinetic and magnetic energies, (b) the kinetic and magnetic enstrophy, and (c) the kinetic palinstrophy — normalized to peak value.

decreasing. In MHD, however, we see an immediate strong energy exchange from kinetic to magnetic which is independent of transport coefficients (Case A and B curves overlay in Fig. 9a for  $t < 10K$ ) with a rapid rise in the magnetic enstrophy (i.e., mean square current). For  $10K < t < 20K$ , there is a flattening in the kinetic energy decay (Fig. 9a) and a subsequent increase in the kinetic enstrophy (Fig. 9b), somewhat akin to Navier-Stokes turbulence. The strength of the respective transport coefficient dictates which particular enstrophy peaks at a greater value (i.e., for Case B the lower resistivity and higher viscosity dictate that the magnetic enstrophy has a greater increase than in Case A while the kinetic enstrophy has a lower increase than in Case A). This is also seen in the sharp rise of the kinetic palinstrophy, Fig. 9c.

The directional energy spectra are shown in Fig. 10 (low magnetic Prandtl number,  $Pr = \nu/\eta = 0.3$ ) and Fig. 11 (high magnetic Prandtl number,  $Pr = \nu/\eta = 3.0$ ). Initially these

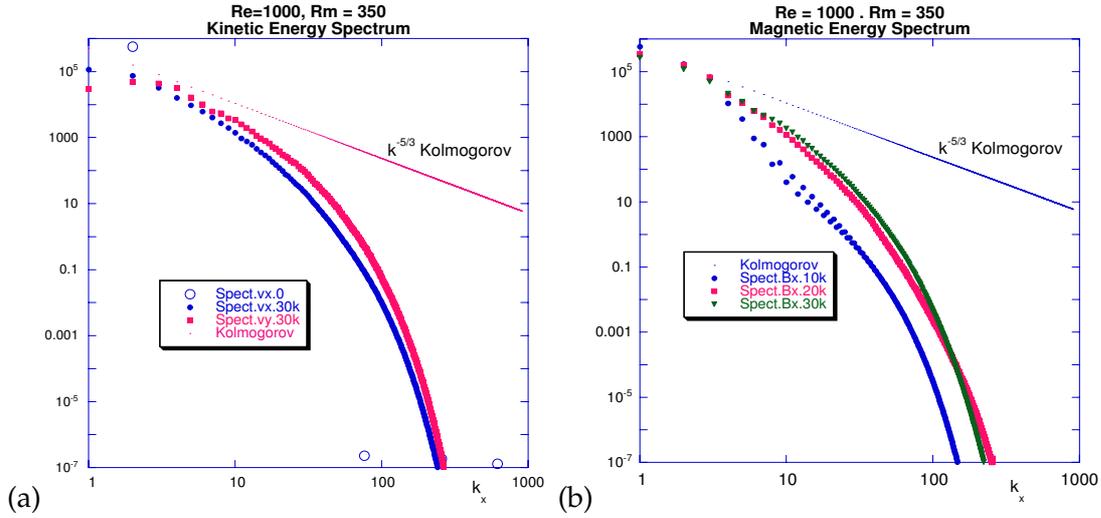


Figure 10: The time development of the directional energy spectra for the low magnetic Prandtl number case:  $Pr = \nu/\eta = 0.3$  (a) the longitudinal and transverse kinetic energy spectrum at  $t = 30K$ , and (b) the longitudinal magnetic energy spectrum at  $t = 10K$ ,  $20K$  and  $30K$ . Also plotted is the Kolmogorov  $k^{-5/3}$  inertial range spectrum.

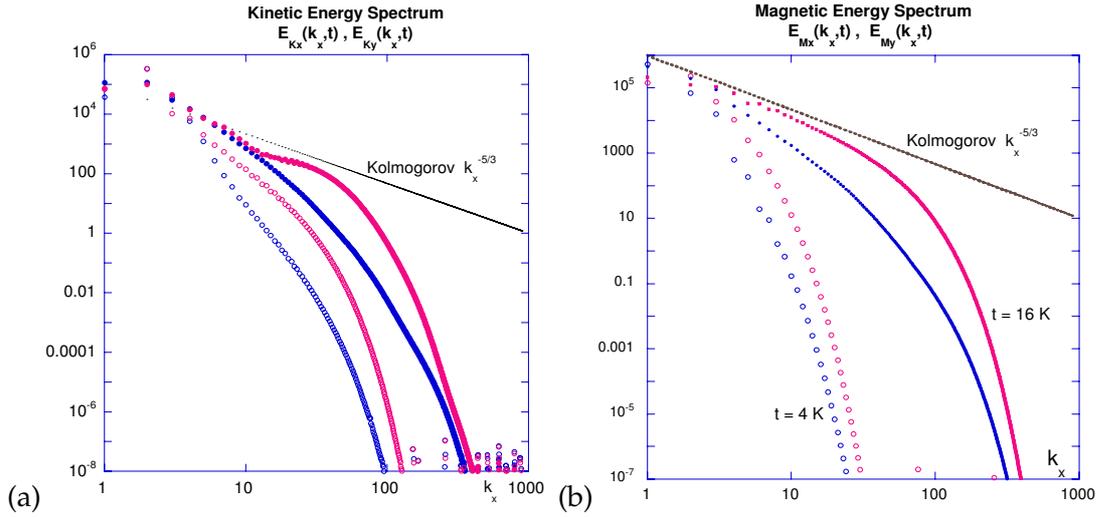


Figure 11: The time development of the directional energy spectra for the high magnetic Prandtl number case:  $Pr = \nu/\eta = 3.0$  (a) the longitudinal and transverse kinetic energy spectrum at  $t = 4K$  and  $t = 16K$ , and (b) the longitudinal and transverse magnetic kinetic energy spectrum at  $t = 4K$  and  $t = 16K$ . Also plotted is the Kolmogorov  $k^{-5/3}$  inertial range spectrum.

spectra are delta functions. The directional kinetic and magnetic spectra are defined by

$$E_{Kx}(k_x, t) = \sum_{k_y, k_z} |u_x(k_x, k_y, k_z, t)|^2, \quad E_{Ky}(k_x, t) = \sum_{k_y, k_z} |u_y(k_x, k_y, k_z, t)|^2, \quad (4.11)$$

$$E_{Mx}(k_x, t) = \sum_{k_y, k_z} |B_x(k_x, k_y, k_z, t)|^2, \quad E_{My}(k_x, t) = \sum_{k_y, k_z} |B_y(k_x, k_y, k_z, t)|^2, \quad (4.12)$$

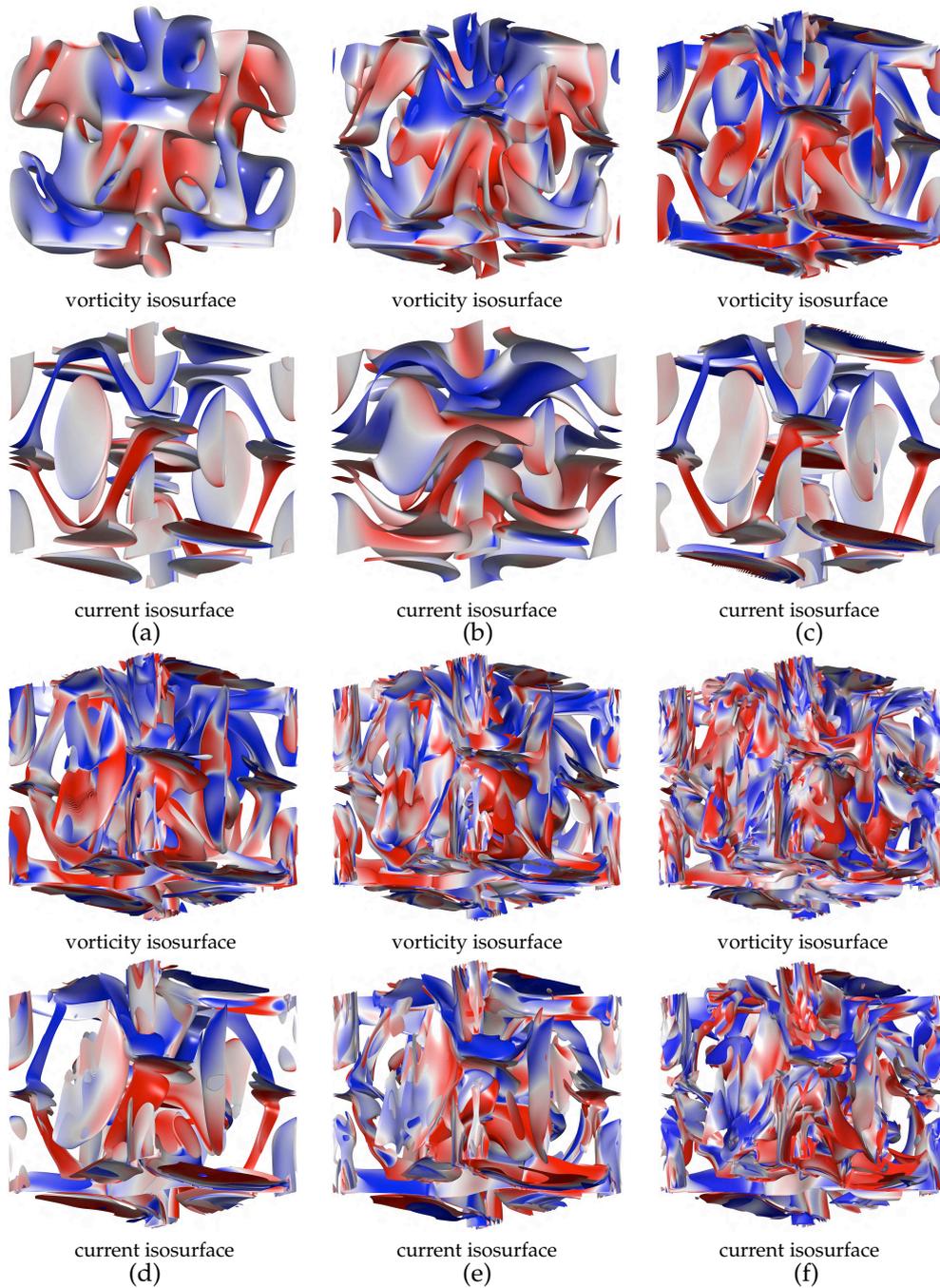


Figure 12: The time development of the vorticity,  $|\omega|$ , and corresponding current  $|\mathbf{J}|$ , isosurface for the high magnetic Prandtl number simulation. (a)  $t=4K$ , (b)  $t=8K$ , (c)  $t=12K$ , (d)  $t=16K$ , (e)  $t=20K$ , (f)  $t=24K$ . The isosurface value chosen is that corresponding to the average  $|\omega|$  and average  $|\mathbf{J}|$  for that time instant. The color coding is dependent on the value of  $\hat{\mathbf{u}} \cdot \hat{\boldsymbol{\omega}}$  and  $\hat{\mathbf{B}} \cdot \hat{\mathbf{j}}$  at the isosurface gridpoint, going from RED for parallel unit vectors  $\hat{\mathbf{u}} \cdot \hat{\boldsymbol{\omega}}=1$  to BLUE for antiparallel unit vectors  $\hat{\mathbf{u}} \cdot \hat{\boldsymbol{\omega}}=-1$ . Similarly for the current isosurfaces: from RED for  $\hat{\mathbf{B}} \cdot \hat{\mathbf{j}}=+1$  to BLUE for  $\hat{\mathbf{B}} \cdot \hat{\mathbf{j}}=-1$ . The GREY scale is for isosurfaces with  $\hat{\mathbf{u}} \cdot \hat{\boldsymbol{\omega}}=0=\hat{\mathbf{B}} \cdot \hat{\mathbf{j}}$ .

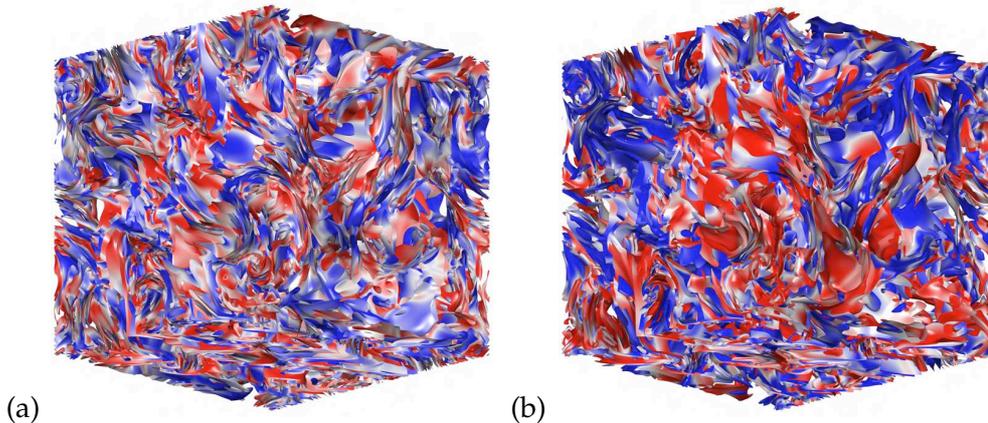


Figure 13: The (a) vorticity and (b) current isosurfaces at  $t=50K$  for the low Prandtl number ( $Pr=\nu/\eta=0.3$ ) simulation. Some large scale magnetic structures persist, along with corresponding large scale vortex structures.

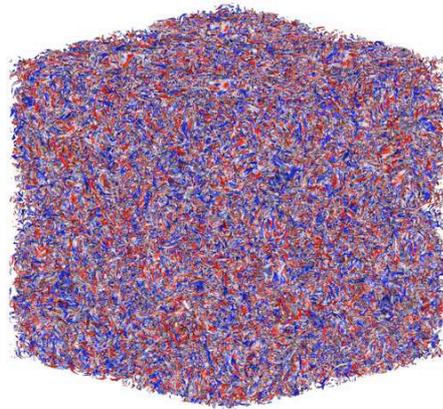


Figure 14: The vorticity isosurfaces in 3D Navier-Stokes turbulence. The flow is dominated by very small scale structures after the inviscid vortex stretching and the peak in the fluid enstrophy. This isosurface is at  $t=7K$  of the ELB simulation discussed in Section 4.1 [15].

where the summation is always over the wavenumbers  $k_y, k_z$ : the longitudinal spectra involve the  $x$ -component of the fields while the transverse spectra the  $y$ -component of the fields. In Fig. 10a, we plot the longitudinal [initially a  $\delta(k_x-2)$ -spectrum] and the transverse [initially a two-delta function peak spectrum at  $k_x=2,4$ ] directional kinetic energy at time = 30K, while in Fig. 10b the directional longitudinal magnetic energy spectrum at  $t=10K, 20K$ , and 30K — and the comparison to the  $k^{-5/3}$  Kolmogorov spectrum.

As we increase the magnetic Prandtl number to Case B one finds a substantial difference between the longitudinal and transverse spectra, Fig. 11. The transverse kinetic and magnetic energy spectra show much stronger excitation of high  $k_x$ -modes, which in the magnetic energy case shows a quite strong semblance to the Kolmogorov inertial energy  $k_x^{-5/3}$ -spectrum at  $t=16K$  (Fig. 11b). There is also an interesting enhancement of the

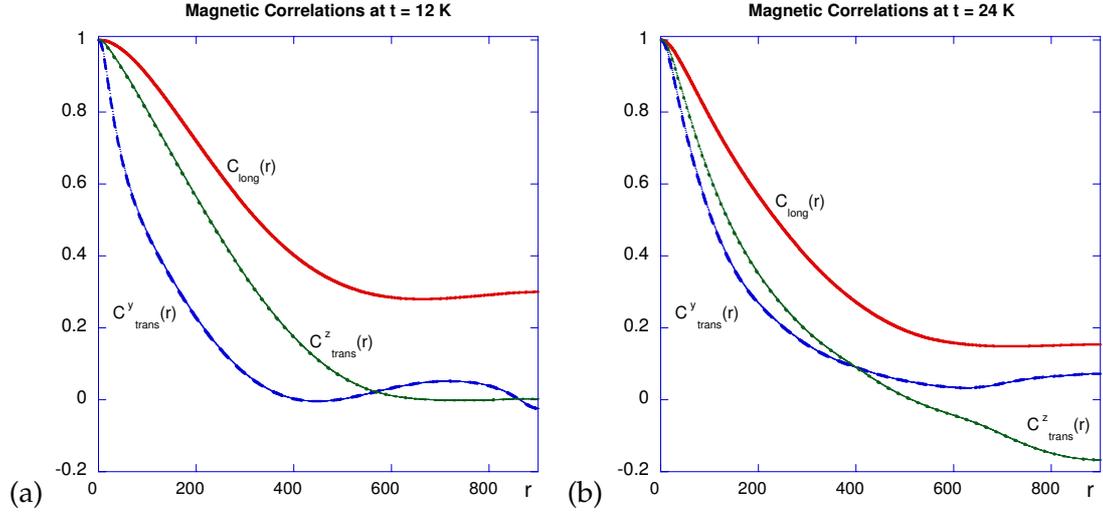


Figure 15: Magnetic Correlations at (a)  $t=12\text{K}$ , (b)  $t=24\text{K}$  for the high magnetic Prandtl number simulation, Case B. The very slight increase in the longitudinal correlation function for  $r > 650$  at  $t=12\text{K}$  is no longer present at  $t=24\text{K}$ .

transverse kinetic energy spectrum for  $40 < k_x < 200$  at  $t=16\text{K}$  (Fig. 11a).

This behavior maybe attributed to the strong but localized vorticity and current sheets developing due to the turbulence. In Fig. 12 we plot some time snapshots of the isosurfaces of vorticity and current for this high magnetic Prandtl number case.

There is much information in Fig. 12: the intensification of localized horizontal current sheets (see Fig. 12a-c, midway at the vertical cube edges), the development of intense vertical localized patches of vorticity and current at later times with similar isosurface geometrical structures of vorticity and current. It is also very apparent that large scale magnetic (and hence velocity) structures persist for long times, Fig. 12f. This is also seen in the low magnetic Prandtl number simulation, Fig. 13, where some large scale vorticity and current isosurface structures persist, even at  $t=50\text{K}$ . This is very unlike 3D Navier-Stokes turbulence which is dominated by small scale vortex structures as seen in Fig. 14.

Finally, we present some correlation data for the magnetic field

$$\begin{aligned} C_{long}(r) &= \langle B_x(x,y,z) B_x(x+r,y,z) \rangle, \\ C_{trans}^y(r) &= \langle B_y(x,y,z) B_y(x+r,y,z) \rangle, \\ C_{trans}^z(r) &= \langle B_z(x,y,z) B_z(x+r,y,z) \rangle. \end{aligned} \quad (4.13)$$

It is seen that as time develops, the almost constant asymptotic tail of the longitudinal magnetic correlation function disappears, and by  $t=28\text{K}$ ,  $C'_{long}(r) < 0$  for all  $r$ . Moreover, we find (for  $r > 0$ )

$$C_{trans}^y(r), C_{trans}^z(r) < C_{long}(r), \quad \text{for all times.} \quad (4.14)$$

Eq. (4.12), with  $C'_{long}(r) < 0$ , is consistent with the correlation statistics of a random solenoidal vector field [19].

## 5 Summary and conclusions

We have developed a 3D LB-MHD algorithm that is ideally parallelized and presented some simulation results on a  $1800^3$ -spatial grid that shows the persistence of large scale magnetic and vorticity structures for long times. Moreover, the time development of the correlation statistics of the magnetic field indicate that the  $\mathbf{B}$ -field is becoming more and more random. An important feature of the LB-MHD approach is that the algorithm automatically ensures  $\nabla \cdot \mathbf{B} = 0$  to machine accuracy.

The straightforward LB algorithm, while simple and explicit, suffers from numerical instabilities as  $Re \rightarrow \infty$ ,  $Rm \rightarrow \infty$ . This places upper bounds on the attainable transport coefficients. At the Navier-Stokes level, entropic algorithms [3–11, 15] have been developed that remain unconditionally stable for arbitrary small viscosities. Indeed, we have presented here the first large scale ELB simulations on a  $1600^3$ -grid at  $Re = 25000$ . While our ELB code runs successfully for much higher Reynolds numbers, the turbulence is no longer fully resolved on these ‘small’ grids, and so these results are not presented here. We are currently developing entropic LB-MHD algorithms that would permit simulations at arbitrary small transport coefficients.

While the simulations reported here are on a simple 3D periodic domain, LB algorithms can handle arbitrary geometries [20] without losing their intrinsic parallelization. Nonuniform spatial grids can be readily handled. In these cases, the spatial grid and the kinetic velocity lattice will now no longer overlay. As a result, the streaming step of the LB algorithm will no longer give immediate data at the spatial nodes. One would then resort to interpolation methods to get the streamed information onto the spatial nodes. Moreover all the latest CFD methods for handling arbitrary spatial grid geometries can be immediately brought over to LB. It remains to be seen what price will need to be paid on the parallelization of such augmented LB codes.

Finally, we comment on another interesting aspect of ELB algorithms. The ELB-viscosity  $v_{eff}(\mathbf{x}, t)$ , Eq. (2.17), gives the appearance of an eddy viscosity and immediately raises the question of whether there is any connection between ELB and the Large Eddy Simulations (LES) in turbulence modeling. The simplest LES model is the Smagorinsky [21] model in which the subgrid scales are modeled by an eddy viscosity that is related to the mean rate of strain velocity tensor:

$$v_{smag}(\mathbf{x}, t) = (C_s \Delta)^2 \sqrt{S_{ij} S_{ij}}, \quad (5.1)$$

where the rate of strain tensor (of the resolvable scales)

$$S_{ij} = \frac{1}{2} \left( \frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} \right), \quad (5.2)$$

$\Delta$  is the filter width (defined in the filtering function that separates the resolvable from the subgrid scales) and  $C_s$  is some empirical constant. Obviously, the connection (if any) between the ELB and LES transport coefficients is not obvious: ELB deals with entropy

surfaces and the determination of the collision parameter  $\gamma(\mathbf{x}, t)$  that enforces detailed balance on the pre- and post-collision distribution functions, while LES deals with the rate of strain tensor. It is of much interest that one can immediately construct local LB-LES models [22, 23] that recover the Smagorinsky-CFD LES model. This is because the local strain tensor can be recovered from the second moment of the non-equilibrium distribution function

$$S_{ij} = \frac{1}{2} \left( \frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} \right) = -\frac{3}{2\rho\tau_u} \sum_{\alpha} e_{\alpha i} e_{\alpha j} [f_{\alpha} - f_{\alpha}^{eq}]. \quad (5.3)$$

Of course, this is exactly how  $\nabla \cdot \mathbf{B}$  is recovered from the trace of the *first* moment of the nonequilibrium magnetic distribution function, Eq. (4.7), and by making this first moment antisymmetric we enforce  $\nabla \cdot \mathbf{B} = 0$  to better than  $\mathcal{O}(10^{-15})$ . This also opens up the possibility of examining LES LB-MHD algorithms being developed for CFD techniques by Carati et al. [24–26], where the LB version will be, unlike the CFD code, ideally parallelized. These LES LB-MHD codes are currently being developed.

## Acknowledgments

The authors are grateful for access to the supercomputers at *Earth Simulator* (Japan), NAVO MSRC (DoD), ASC MSRC (DoD), AFRL MSRC (DoD) and NERSC (DoE). The authors were supported by grants from DoE, AFOSR and AFRL as well as the Director, Office of Science, Office of Advanced Scientific Computing Research, Department of Energy under Contract No. DE-AC02-05CH11231.

## References

- [1] J. Carter, M. Soe, L. Oliker, Y. Tsuda, G. Vahala, L. Vahala and A. Macnab, Magnetohydrodynamics turbulence simulations on the Earth Simulator using the lattice Boltzmann method, International Conf. on SuperComputing, SC|05 (Nov. 2005), Gordon-Bell Finalist, Paper ISBN #1-59593-061-2; G. Vahala, J. Carter, M. Soe, J. Yepez, L. Vahala and A. Macnab, Performance of lattice Boltzmann codes for Navier-Stokes and MHD turbulence on high-end computer architectures, in *Parallel CFD 2005*, ed. A. Deane et al., Elsevier, 2006.
- [2] P. J. Dellar, Lattice kinetic schemes for magnetohydrodynamics, *J. Comput. Phys.* **179**, (2002) 95-126
- [3] S. Succi, *The Lattice Boltzmann Equation for Fluid Dynamics and Beyond*, Clarendon Press, 2001 - and references therein.
- [4] S. Chen and G.D. Doolen, Lattice Boltzmann method for fluid flows, *Annu. Rev. Fluid Mech.* **30** (1998) 329–364
- [5] I. V. Karlin, A. Gordan, S. Succi and V. Boffi, Maximum entropy principle for lattice kinetic equations, *Phys. Rev. Lett.* **81** (1998) 6-9.
- [6] S. Ansumali and I. V. Karlin, Stabilization of the lattice Boltzmann method by the *H*-theorem, *Phys. Rev E* **62** (2000) 7999-8003.
- [7] S. Ansumali and I. V. Karlin, Single relaxation time model for entropic lattice Boltzmann methods, *Phys. Rev. E* **65** (2002) 056312

- [8] S. Ansumali, I. V. Karlin and H.C. Ottinger, Minimal entropic kinetic models for simulating hydrodynamics, *Europhy. Lett.* 63 (2003) 798-804.
- [9] A. J. Wagner, An  $H$ -theorem for the lattice Boltzmann approach to hydrodynamics, *Europhys. Lett.* 44 (1998) 144-149.
- [10] B. M. Boghosian, J. Yepez, P. V. Coveney and A. J. Wagner, Entropic lattice Boltzmann methods, *Proc. Roy. Soc. Lond. A457* (2001) 717-766; B.M. Boghosian, P.J. Love, P.V. Coveney, I.V. Karlin, S. Succi, and J. Yepez, Galilean-invariant lattice-Boltzmann models with H theorem, *Phys. Rev. E* 68, 025103 (2003).
- [11] B. Keating, G. Vahala, J. Yepez, M. Soe and L. Vahala, Entropic lattice Boltzmann representations required to recover Navier-Stokes flows, *Phys. Rev. E* 75 (2007) 036712.
- [12] Y. Kaneda, T. Ishihara, M. Yokokawa, K. Itakura and A. Uno, Energy dissipation rate and energy spectrum in high resolution direct numerical simulations of turbulence in a periodic box, *Phys. Fluids* 15 (2003) L21-24.
- [13] G. Wellein, T. Zeiser, S. Donath and G. Hager, On the single processor performance of simple lattice Boltzmann kernels, *Computers & Fluids* 35 (2006) 910-919
- [14] B. Palmer and J. Nieplocha, Efficient algorithms for ghost cell updates on two classes of MPP architectures, *Proc. Parallel Dist. Comp. Systems '02*, (2002) 192
- [15] G. Vahala, J. Yepez, M. Soe, L. Vahala and S. Ziegeler, Lattice Boltzmann Algorithms for Fluid Turbulence, *hpcmp-ugc*, pp. 52-56, HPCMP Users Group Conference 2007, 2007.
- [16] S. Kida and Y. Murakami, Kolmogorov similarity in free decaying turbulence, *Phys. Fluids* 30 (1987) 2030-2039.
- [17] G. Breyiannius and D. Valougeorgis, Lattice kinetic simulations in three-dimensional magnetohydrodynamics, *Phys. Rev. E* 69 (2004) 065702(R).
- [18] A. I. D. Macnab, G. Vahala, L. Vahala, J. Carter, M. Soe and W. Dorland, Non-local closure and parallel performance of lattice Boltzmann models for some plasma physics problems, *Physica A* 362 (2006) 48-56.
- [19] S. Panchev, *Random Functions and Turbulence*, Pergamon Press, 1971.
- [20] H. Chen, Volumetric formulation of the lattice Boltzmann method for fluid dynamics: basic concepts, *Phys. Rev. E* 58 (1998) 3955-3963; H. Fan, R. Zhang and H. Chen, Extended volumetric scheme for lattice Boltzmann models, *Phys. Rev. E* 73 (2006) 066708
- [21] J. Smagorinsky, General circulation experiments with the primitive equations: I. the basic equations, *Mon. Weather Rev.* 91 (1963) 99-164.
- [22] S. Hou, J. Sterling, S. Chen and G.D. Doolen, A lattice Boltzmann subgrid model for high Reynolds number flows, in: *Pattern Formation and Lattice Gas Automata*, A.T. Lawniczak, R. Kapral (Eds.), *Fields Inst. Comm.*, 6 (1996) 151-166.
- [23] H. Yu, S. S. Girimaji and L-S. Luo, DNS and LES of decaying isotropic turbulence with and without frame rotation using lattice Boltzmann method, *J. Computat. Phys.* 209 (2005) 599-616.
- [24] O. Aguilo, W-C. Muller, B. Knaepen and D. Carati, LES of decaying MHD turbulence with dynamic subgrid-modeling, *Phys. Plasmas* 8 (2001) 3502-3505.
- [25] W-C. Muller and D. Carati, LES of MHD turbulence, *Comp. Phys. Comm.* 147 (2002) 544-547.
- [26] W-C. Muller and D. Carati, Dynamic gradient-diffusion subgrid models for incompressible MHD turbulence, *Phys. Plasmas* 9 (2002) 824-834.