Yin-Yang-Zhong grid: An overset grid system for a sphere

Hayashi, Hiroshi

Kageyama, Akira

(Citation)
Journal of Computational Physics,305:895-905

(Issue Date)
2016-01-15

(Resource Type)
journal article

(Version)
Accepted Manuscript

(URL)
https://hdl.handle.net/20.500.14094/90003497

Yin–Yang–Zhong Grid: An Overset Grid System for a Sphere

Hiroshi Hayashi and Akira Kageyama*

Department of Computational Science,
Kobe University,
Kobe 657-8501, Japan

Abstract

For numerical simulations inside a sphere, an overset grid system, Yin–Yang–Zhong grid, is proposed. The Yin–Yang–Zhong grid is an extension of the Yin–Yang grid, which is widely used in various simulations in spherical shell geometry. The Yin–Yang grid is itself an overset grid system with two component grids, and a new component grid called Zhong is placed at the center of the Yin–Yang grid. The Zhong grid component is constructed on Cartesian coordinates. Parallelization is intrinsically embedded in the Yin–Yang–Zhong grid system because the Zhong grid points are defined with cuboid blocks that are decomposed domains for parallelization. The computational efficiency approaches the optimum as the process number increases. Quantitative test simulations are performed for a diffusion problem in a sphere with the Yin–Yang–Zhong grid. Correct decay rates are obtained by the simulations. Two other tests in magnetohydrodynamics (MHD) in a sphere are also performed. One is an MHD dynamo simulation, and the other is an MHD relaxation simulation in a sphere.

1 Introduction

Numerical simulations inside a sphere, or a ball, are required in a wide variety of problems in physics, such as spherical turbulence generators [1], spherical reaction–diffusion problems [2], magnetohydrodynamics (MHD) in a ball [3], and melting problems inside a spherical container [4]. Simulations in spherical geometry are particularly important in geophysics and astrophysics.

A sphere is a workbench for developers of numerical schemes. Various methods have been applied in the spatial discretization of a ball region, from the finite element method [5, 6, 7], to the finite difference on a special coordinate system [8] and the spectral element method [9, 10]. Pseudo-spectral methods are

*kage@port.kobe-u.ac.jp

an important class of such methods, which are based on the spherical harmonics expansion in the horizontal (latitudinal and longitudinal) directions, that can be further classified into subgroups by the adopted radial discretization method; expansion by the spherical Bessel functions [11], the Chebyshev polynomials [12], and the Zernike polynomials [13]. The finite difference method in the radial direction is also used [14].

The finite difference/volume methods in all directions are attractive approaches for massively parallel computations. However, constructing a structured grid inside a ball is not simple because there is no orthogonal coordinate system that fits a ball without a coordinate singularity. The spherical polar coordinate system $\{r, \theta, \phi\}$, where r, θ, and ϕ denote the radius, colatitude, and longitude, respectively, has two types of coordinate singularities: One is on the poles ($\theta = 0$ and π), and the other is at the origin $r = 0$. The coordinate singularity itself is not a problem, in general, because one can always convert the equations into non-singular forms by applying L'Hospital's rule. For example, non-singular forms of the MHD equations on the poles ($\theta = 0, \pi$) are derived and solved by a finite difference method in [15]. Serious numerical problems are observed in the grids near the poles, rather than just on the poles. The concentrated grids around a coordinate singularity drastically reduce the time step when an explicit time integration scheme is adopted. Even when an implicit scheme is used, an unbalanced distribution of the grids damages the computational efficiency because there is generally no physical reason to place a very fine grid mesh near a coordinate singularity.

To avoid the grid convergence near the north and south poles, we proposed the Yin–Yang grid [16], which is an overset grid system applied to the spherical surface or the spherical shell geometry between two concentric spheres. The Yin–Yang grid is now successfully applied in geophysics [17, 18, 19, 20], planetary/space physics [21, 22, 23], solar physics [24, 25, 26], astrophysics [27, 28, 29, 30], and image processing [31].

Since the basic idea of the Yin–Yang grid is to use the low latitude part of the spherical polar coordinates to avoid the grid concentration near $\theta = 0$ and π, another grid concentration near $r = 0$ is left. Therefore, in most cases, simulations using the Yin–Yang grid have a spherical cavity at the center. It is possible to apply the Yin–Yang grid for a ball region including the origin. In such a case, however, one has to overcome the severe restriction on the time step due to the concentrated grid points near $r = 0$ when an explicit time integration scheme is used. In this paper, we propose placing another grid element for the overset grid at the center of the sphere. The new grid element is called Zhong, which means "center" in Chinese. We call the whole overset grid system "Yin–Yang–Zhong", which is shown in Fig. 1.

In this study, we focus on the grid arrangement of the Yin–Yang–Zhong and on its parallel implementation. In test applications of this grid system (Section 4), we use the second-order central difference and the trilinear interpolations between Yin–Yang and Zhong. The bilinear interpolations are applied between Yin and Yang because they share same radii. It is straightforward to improve the numerical accuracy by using the general theory of the overset grid

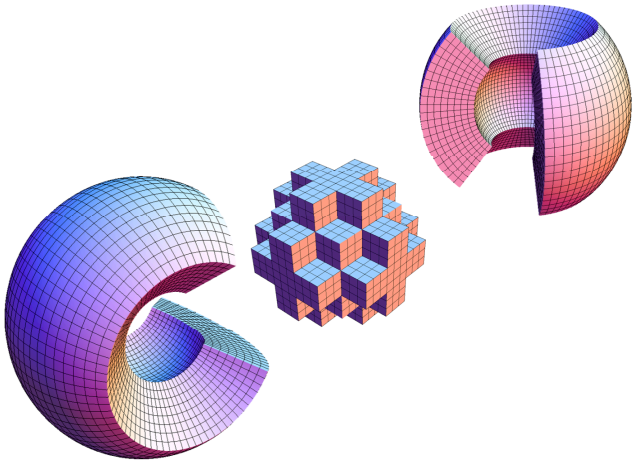

Figure 1: An overview of the Yin–Yang–Zhong grid. The two congruent elements, Yin and Yang, are combined to form a Yin–Yang grid. The central component is a set of cuboid blocks of Cartesian grids in which the Zhong component grids are defined to cover the central cavity of the Yin–Yang grid with minimum overlap.

method [32]. When special attention needs to be paid to the conservation laws for the interpolations, various techniques such as [33], which is successfully applied to the Yin–Yang grid, can be used in the Yin–Yang–Zhong grid. Schemes other than the finite difference/volume methods, such as the spectral Galerkin method, which is successfully used in the Yin–Yang grid [34], can also be applied to the Yin–Yang–Zhong grid.

2 Yin–Yang–Zhong Grid

The Yin–Yang–Zhong grid is an overset grid system that fits a ball region. In this study, for the sake of simplicity, we assume that its radius is $r = 1$. The Yin–Yang–Zhong grid is divided into two parts. The outer spherical shell region $r_c \leq r \leq 1$ is mainly covered by the Yin–Yang grid system and partially covered by the Zhong grid system, while the inner sphere of $0 \leq r \leq r_c$ is covered by the Zhong grid system. We call the $0 \leq r \leq r_c$ region "core" and the $r_c \leq r \leq 1$ region "shell." The core radius r_c is a free parameter in the Yin–Yang–Zhong grid method.

2.1 Shell elements: Yin and Yang

The shell region is discretized with the Yin–Yang grid, which is itself an overset grid system with two congruent component grids, Yin and Yang [16]. We arrange the spherical polar coordinates $\{r, \theta, \phi\}$ and the Cartesian coordinates $\{x, y, z\}$

in such a way that $\theta = 0$ is on the $+z$ axis and $\phi = 0$ is on the $+x$ axis.

The Yin component spans a low latitude region in the spherical polar coordinates as follows:

$$\{r_{\rm c} - \delta \le r \le 1,\ \theta_- \le \theta \le \theta_+,\ \phi_- \le \phi \le \phi_+\}\,, \tag{1}$$

where $\theta_\pm = \pi/2 \pm (\pi/4 + \delta)$, $\phi_\pm = \pm(3\pi/4 + \delta)$. The angle δ is a small buffer that is necessary for mutual interpolations in the overset grid method. The minimum value of δ depends on the grid size and the order of the adopted scheme [32]. The Yang component is defined by the same equation (1) under a set of rotated Cartesian axes $\{x', y', z'\}$ with $\{x', y', z'\} = \{-x, z, y\}$. The Yin and Yang are combined with partial overlap near their borders. Following the standard overset grid methodology [32], we apply mutual interpolations on the boundary grids of the Yin and Yang component grids. As for the radial grid points, here, we assume a uniform grid distribution with the spacing Δr, although one may take any stretched (non-uniform) grid system for the r direction in general.

2.2 Core element: Zhong

To construct the Zhong grid, we first consider a cube that encloses the core. We call this cube "l-cube" ("l" stands for large). The l-cube circumscribes a sphere R_{c+} of radius $r = r_{\rm c} + \delta$. See the green square in Fig. 2. The edges of the l-cube are parallel to the x–y–z axes. The Zhong component grid is defined on the basis of a Cartesian grid system with the same uniform spacing in the x, y, and z directions. Here, for the sake of simplicity, we further assume that this spacing is the same as the radial grid spacing of the Yin–Yang grid in the shell: $\Delta x = \Delta y = \Delta z = \Delta r$.

While the overset grid methodology allows us to compute the core region with all the Cartesian grid points in the l-cube, such a grid arrangement for Zhong is obviously a waste of memory and computational time: Even if δ is negligibly small, the volume ratio $\Gamma_{\rm v}$ of the inscribed core (radius: $r_{\rm c}$) and the cube (edge length: $2r_{\rm c}$) is $\Gamma_{\rm v} = \pi/6 \sim 1.91$. In other words, it is sufficient to keep only about a half of the Cartesian grid points in the l-cube as Zhong's grid points and the other half can be discarded. To realize an optimized computation with the Zhong grid on parallel computers, we separately consider the computational efficiency and the memory efficiency in the following.

To maximize the computational efficiency, we define Zhong's grid points as a set of (i) grid points that are located inside the sphere R_{c+} and (ii) grid points that are located just outside R_{c+} in the sense that it has at least one neighbor grid point in the sphere R_{c+}. The black mesh in the right panel in Fig. 2 shows a Zhong grid defined above.

The computational waste is measured by the volume ratio $\Gamma_{\rm v}$ of Zhong's cells and the core's sphere. Figure 3 shows $\Gamma_{\rm v}$ as a function C_x, where C_x denotes the cell number of the l-cube in the x direction (The grid number in x is $L_x = C_x + 1$.). Since we are assuming that $C_x = C_y = C_z$, the total cell number is $C_x \times C_y \times C_z = C_x^3$ and is plotted on the upper horizontal axis

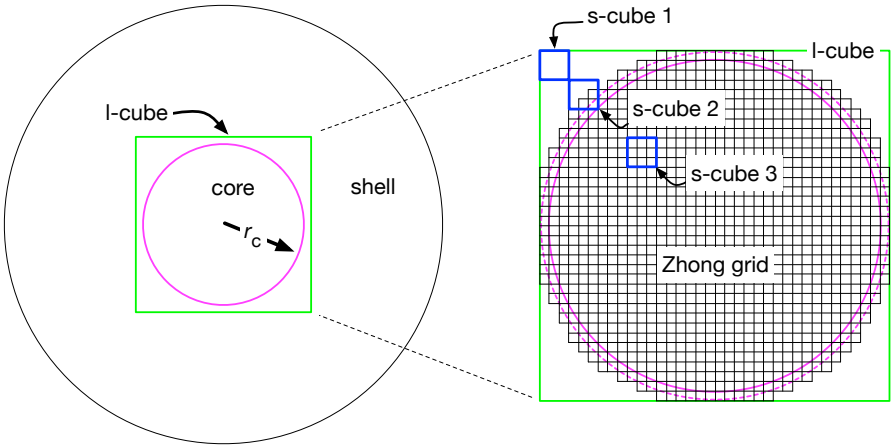

Figure 2: Construction of the Zhong component grid. The sphere of radius $r = r_c$ is called core and its outer part ($r > r_c$) is called shell. l-cube is a cube that circumscribes a sphere R_{c+} of radius $r = r_c + \delta$ (magenta dashed circle on the right). The Zhong grid (black mesh in the right) is a Cartesian-based uniform grid system defined in the l-cube with minimum overlap with the shell. The whole l-cube is divided into multiple cuboids, called s-cubes for parallel processing. An MPI process is in charge of an s-cube. In the right panel, grid points in s-cube 3 are fully inside the sphere of $r = r_c + \delta$ and all the grid points in this s-cube are used in the computations. S-cube 2 is partially outside the sphere. The memory for all grid points in s-cube 2 are allocated in the MPI process for this s-cube, but the grid points located outside the sphere are not used in the computations. S-cube 1 is fully outside the sphere, and no MPI process is assigned to this s-cube from the first place.

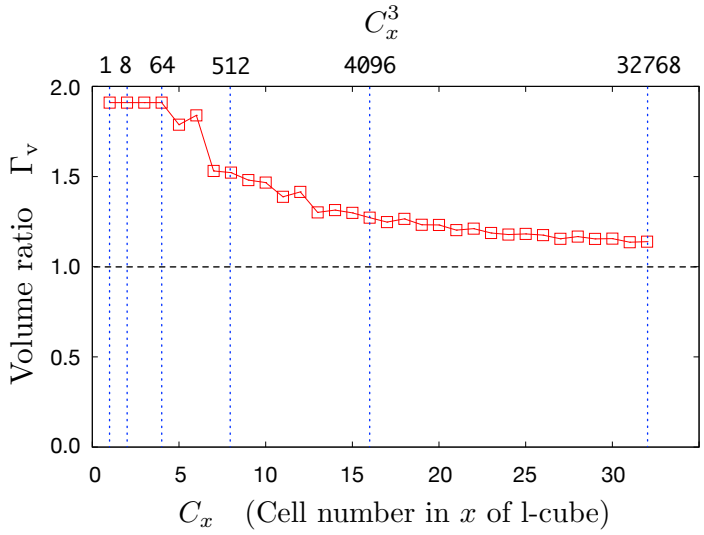

Figure 3: Volume ratio Γ_v of Zhong's grid cells and the core sphere, which is regarded as a kind of measure for the computational waste of the Zhong grid. As the cell number for the division of the l-cube increases, Γ_v approaches the minimum value 1.

in Fig. 3. As the cell number increases, the volume ratio Γ_v approaches the optimum value 1. Γ_v starts decreasing from the worst value $\pi/6 \sim 1.91$ at $C_x = 5$, which is too coarse for practical simulations, and when $C_x = 32$, or when the total cell number $C_x^3 = 32,768$, $\Gamma_v = 1.14$, which would be acceptable.

As the total cell number C_x^3 increases, the profile of Zhong's grid points approaches the sphere. However, the irregular boundary profile is not desirable from the point of view of the memory allocation in simulation codes. Since the most efficient arrangement of grid points for the memory allocation is a three-dimensional rectangular box, or cuboid, we allocate the memory for the Zhong grid as an assembly of cuboid regions in the following manner: We note that this construction is a natural consequence of the domain decomposition of the l-cube, which is inevitable in parallel simulations.

We divide the l-cube into $P_x \times P_y \times P_z$ pieces, where P_x, P_y, and P_z denote the division number, or process number, in the x, y, and z directions, respectively. We call the small piece of the cuboid "s-cube". Each s-cube is supposed to be executed by an MPI process.

In the right panel of Fig. 2, three sample s-cubes are denoted by blue squares. Among them, the location of s-cube 3 is fully inside the core. All the grid points (black mesh) in s-cube 3 are used for the computation. S-cube 2, on the other hand, is partially outside the sphere. Only some of the grid points in s-cube 2 is used in the computation; the white part on the upper-left in s-cube 2 is not used in the computation, although the memory for grid points there are allocated in

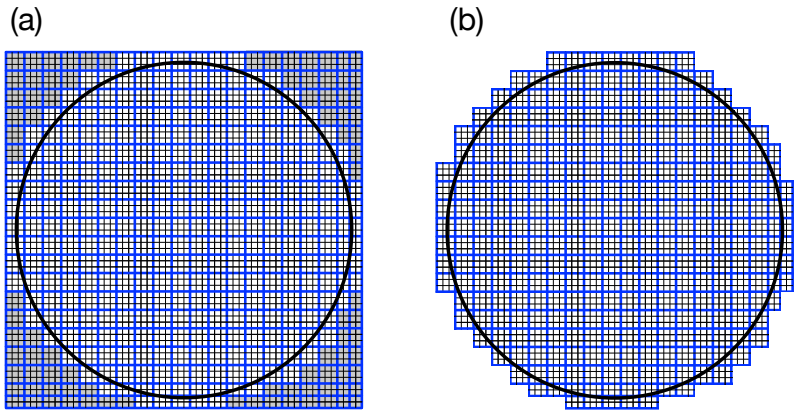

Figure 4: An example of s-cubes. (a) l-cube is divided into multiple cuboid domains, or s-cubes, that are denoted by blue lines. The s-cubes with a gray background are fully outside the sphere R_{c+} of radius $r = r_{\rm c}+\delta$. (b) The s-cubes with a gray background are discarded, and only the remaining s-cubes, which are either fully or partially inside R_{c+}, are included in the execution. An MPI process is allocated for each s-cube. The resulting s-cubes do not necessarily have a spherically symmetric profile.

the MPI process that is in charge of s-cube 2. Finally, s-cube 1, shown in Fig. 2, is fully outside the core and no grid point in this s-cube is used in the computation. No memory, and no process, is assigned to s-cube 1. Therefore, in effect, s-cube 1 does not exist.

We describe the above selection procedure of s-cubes in more detail by using Fig. 4. The blue lines in Fig. 4(a) denote potential s-cubes, some of which will be discarded in the selection. Each s-cube contains uniform grid points (black mesh). The s-cubes that are fully outside of the sphere R_{c+} of radius $r_{\rm c} + \delta$, or the cells with a gray background in Fig. 4(a), are discarded. In other words, no MPI process for these s-cubes is assigned from the beginning of the program's execution. On the other hand, MPI processes are assigned to the s-cubes whose grid points are partially or totally inside R_{c+}. The MPI processes, thus generated, retain the minimum s-cube profiles to enclose the core as shown in Fig. 4(b). Since the cell number $C_x(= C_y = C_z)$ of the l-cube is not always a multiple of process size P_x in general, all s-cubes do not necessarily have the same number of grid points. Therefore, the resulting profile of the s-cubes is not necessarily spherically symmetric, as shown in Fig. 4(b). Even so, the memory waste decreases when the number of domains, or the total number of MPI processes for the Zhong grid, is sufficiently large.

Figures 5(a) and (b) show magnified views of s-cubes that are just on the core surface of $r = r_{\rm c}$, which is denoted by a solid magenta curve. The dashed magenta curves represent the spheres $R_{c\pm}$ of radius $r = r_{\rm c} \pm \delta$. The grid points

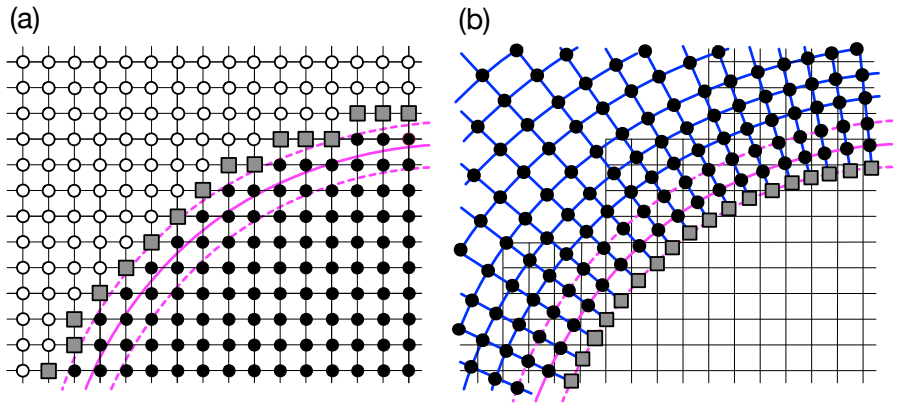

Figure 5: Magnified views of s-cubes that are just on the core–shell border of the sphere R_c of radius $r = r_c$ that is denoted by a solid magenta curve. Dashed magenta curves denote spheres $R_{c\pm}$ with radius $r = r_c \pm \delta$. (a) Grid points of the s-cubes. Variables on gray squares are set by interpolations from the Yin–Yang grid. (b) Grid points of the Yin–Yang grid. Variables on the gray squares are set by interpolations from the s-cubes in the Zhong grid.

with a black circle in Fig. 5(a) are fully inside the sphere R_{c+}. Physical variables on the black circles are updated by the basic equations to be solved. On the other hand, the grid points with white circles in Fig. 5(a) are fully outside of R_{c+}. Physical variables on these circles are not calculated, but arrays for these variables are stored in the MPI process for this s-cube. The variables on grid points with gray squares are just outside of R_{c+}. Physical variables on these squares are set by interpolations from the Yin–Yang grid.

Conversely, the physical variables on the grid points on the inner spherical boundary of the Yin–Yang grid at R_{c-} [grey squares in Fig. 5(b)] are set by interpolations from data on the Zhong grid. The mutual interpolation between the component grids is a key point in the overset grid method.

3 Parallelization

For efficient computations and optimized process allocation, we construct nested communicators of MPI as shown in Fig. 6. MPI processes that are in charge of the Yin (Yang) component grid constitute the Yin (Yang) communicator, and MPI processes for s-cubes or Zhong grids constitute the Core communicator. The members of the Yin communicator and the Yang communicator constitute the Shell communicator.

We apply a two-dimensional domain decomposition in the horizontal directions for the parallelization of the Yin and Yang: We divide the whole Yin (or Yang) region into $P_\theta \times P_\phi$ pieces in the horizontal directions. When $P_\phi = 3\,P_\theta$,

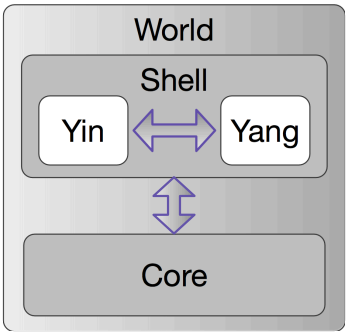

Figure 6: A design of MPI communicators for the Yin–Yang–Zhong grid system. The MPI processes for the Yin (Yang) component grid constitute the Yin (Yang) communicator. The members of the two communicators constitute the Shell communicator. The MPI processes for the core, or the Zhong grid, constitute the Core communicator. The arrows denote the inter-communicator communications.

each divided piece has a regular square, which is desirable for the optimum inter-process communication, in the horizontal space because the domain of ϕ is three times larger than the domain of θ [see eq. (1)]. We can make use of MPI's virtual (two-dimensional Cartesian) topology for Yin and Yang for automatic and optimized allocation of process ranks in the communicators. On the other hand, we allocate the rank numbers in the Core communicator by hand.

We show in Fig. 7 a measured scaling of the Yin–Yang–Zhong grid method with an MHD simulation program used in the relaxation simulation inside a sphere that will be described in Section 4.3. The horizontal axis is the process number P for the Core communicator or for the Zhong grid. The vertical axis denotes the simulation's relative speed normalized by the computational time when the Zhong's process number is $P = 96$. (The relative speed when $P = 96$ is defined as 96.) The process number for the shell part, or the Yin-Yang grid part, is almost the same as that for the Zhong grid in each case. The computer system used is Fujitsu FX–10. The grid number for the Yin–Yang grid is $N_r \times N_\theta \times N_\phi \times \mathrm{Yin/Yang} = 151 \times 154 \times 458 \times 2$, and the grid number for the l-cube is $L_x \times L_y \times L_z = 163 \times 163 \times 163$.

Even with the same process number P for the core region, the relative speed depends on how we divide the l-cube: $P = P_x \times P_y \times P_z$. The relative speed for each P in Fig. 7 is the maximum value measured by multiple runs for each P with six to twelve different P_x, P_y, and P_z combinations, including one-dimensional decomposition ($P_z = P_y = 1$), two-dimensional decompositions ($P_z = 1$), and three-dimensional decompositions. Even though it is not a thorough examination for each process number P, the speedup in Fig. 7 shows an almost linear scaling of up to 960 processes. The maximum number 960 in this plot comes from the computer resource available for this study. The scaling in

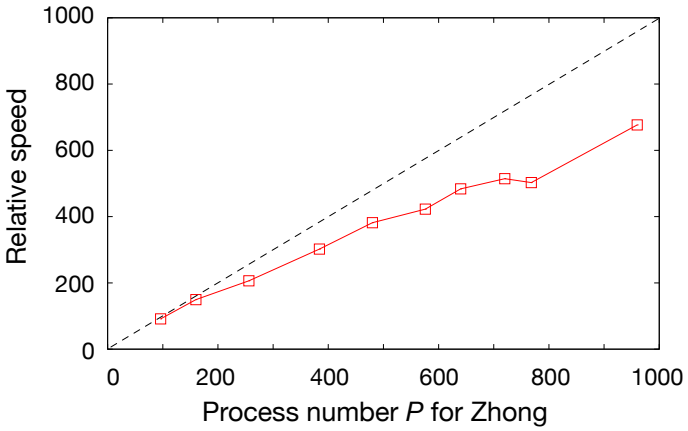

Figure 7: Relative speed of an MHD simulation in a sphere with the Yin–Yang–Zhong grid, as a function of total process number P for the Zhong part.

Fig. 7 suggests that the code runs with more than $1,000$ processors.

4 Tests and Applications

4.1 Diffusion equation

To confirm the accuracy of the Yin–Yang–Zhong grid method, we first solve a diffusion problem inside a sphere and compare numerical and analytical solutions.

In this test, we solve a diffusion equation of a scalar field $T(r, \theta, \phi)$ with a constant diffusion coefficient:

$$\frac{\partial T}{\partial t} = \eta \nabla^2 T, \tag{2}$$

in a sphere of $0 \leq r \leq 1$ with the Dirichlet boundary condition: $T(r = 1) = 0$. When we consider T to be temperature, this initial-boundary-value problem corresponds to a thermal diffusion problem in a ball with a fixed temperature on the surface. For a free decay problem of a magnetic field in a sphere, T corresponds to the toroidal field potential, although we will retain a physically irrelevant degree $\ell = 0$ in the test.

The general solution of this problem is written as [35]

$$T(r, \theta, \phi, t) = \sum_{\ell, n} c_{\ell n} \, \exp(-\lambda_{\ell n} t) \, r^{-1/2} j_{\ell + \frac{1}{2}}(x_{\ell n} r) \, Y_\ell(\theta, \phi), \tag{3}$$

for $n \geq 1$, $\ell \geq 0$, where Y_ℓ denote the spherical harmonics, $j_{\ell+1/2}(r)$ represent the modified Bessel functions, and $c_{\ell n}$ indicate the constants. The damping

rates $\lambda_{\ell n}$ are given by $\lambda_{\ell n} = \eta\, x_{\ell n}^2$, where $x_{\ell n}$ denotes the n-th zero-point of $j_{\ell+1/2}(r)$.

We numerically solve the time development of eq. (2) with the Yin–Yang–Zhong grid from various initial conditions and compared them with the analytical solutions. We apply the second-order central difference method for the spatial discretization and the fourth-order Runge-Kutta method for temporal integration. The mutual interpolations between Yin, Yang, and Zhong are bilinear (between Yin and Yang) or trilinear (between Zhong and Yin–Yang). The core radius $r_{\rm c} = 0.35$. The diffusion coefficient is $\eta = 6.3588 \times 10^{-2}$. The total grid number is $N_r \times N_\theta \times N_\phi \times {\rm Yin/Yang} = 101 \times 104 \times 308 \times 2$ for the Yin–Yang grid part, plus $L_x \times L_y \times L_z = 114 \times 114 \times 114$ for the l-cube (Zhong grid) part. The grid spacing in the radial direction is $\Delta r = 6.5 \times 10^{-3}$. The grid spacings in the Zhong grid, Δx, Δy, and Δz, are the same. We used 96 to 640 processors in these tests.

We first performed a simulation from an initial condition of a random $T(r, \theta, \phi)$ field that was constructed by a superposition of eleven spherical harmonics modes $Y_\ell^m(\theta, \phi)$ for $0 \leq \ell \leq 10$ with randomly chosen m ($|m| \leq \ell$) for each ℓ and arbitrarily chosen phase shifts in ϕ for each m. The initial amplitude of each mode is unity. In the radial direction, we take linear combinations of sinusoidal functions and modified Bessel functions satisfying the boundary condition. We followed the time development of eq. (2) until the time derivative of T converged to a constant in the whole spherical region. The survived mode is the eigenfunction that has the smallest damping rate, and the converged constant of the time derivative is the damping rate. The numerically obtained damping rate thus calculated with the Yin–Yang–Zhong grid is $\lambda = 0.627867$. The analytical value is $\lambda_{01} = \eta\, \pi^2 = 0.627590$ for the eigenvalue pair $(\ell, n) = (0, 1)$. The relative error is 2.77×10^{-4}.

We also compared the numerically obtained decay rates with the analytical values for different eigenvalue pairs (ℓ, n), by performing the diffusion simulations with their eigenfunctions $r^{-1/2} j_{\ell+\frac{1}{2}}(x_{\ell n} r)\, Y_\ell(\theta, \phi)$ as the initial conditions. The results are summarized in Fig. 8. The theoretical values of decay rates $\lambda_{\ell n} = \eta\, x_{\ell n}^2$ for $n = 1$ and 2 are shown in the red and green curves. The numerically obtained values with the Yin–Yang–Zhong grid are plotted by using crosses ($n = 1$) and squares ($n = 2$). The decay rate for $(\ell, n) = (0, 1)$ is the same as that obtained under the randomly set initial condition described above with the error of $\mathcal{O}(10^{-4})$. Other decay rates numerically obtained by using the Yin–Yang–Zhong grid also coincide with the theoretical values with the same order of error.

4.2 MHD dynamo with diffusive core

The second test that we performed is an MHD simulation inside a rotating sphere of constant angular velocity $\boldsymbol{\Omega}$. Considering applications to planetary and stellar dynamo simulations in the future, we assume a convectively unstable MHD layer in the outer spherical shell region ($r_{\rm i} \leq r \leq 1$), and a convectively stable, magnetically diffusive region in the center ($0 \leq r \leq r_{\rm i}$). Although

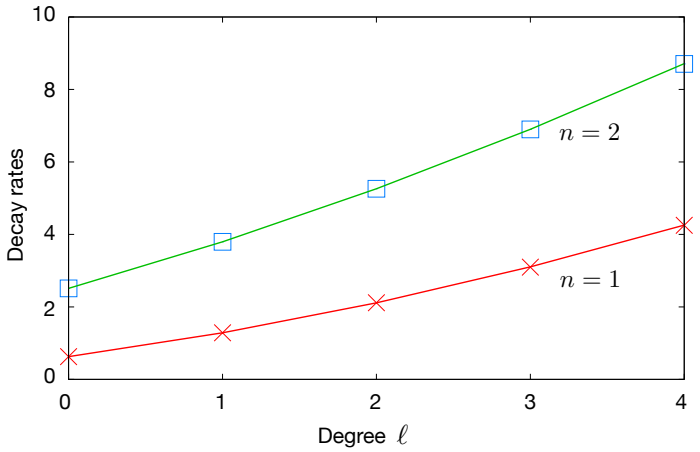

Figure 8: Decay rates of a scalar field caused by diffusion inside a sphere of radius $r = 1$ with the diffusion coefficient $\eta = 6.3588 \times 10^{-2}$. The analytical solution is $\lambda_{\ell n} = \eta\, x_{\ell n}^2$, where $x_{\ell n}$ denotes the n-th zero-point of the spherical Bessel function $j_{\ell+1/2}(x)$. The lower red curve shows the analytical solution for $n = 1$, and the upper green curve is for $n = 2$. The squares and crosses denote the numerical solution obtained by using the Yin–Yang–Zhong grid.

$r_{\rm i}$ does not necessarily have to coincide with $r_{\rm c}$ between the Zhong grid and the Yin–Yang grid, here, we set $r_{\rm i} = r_{\rm c} = 0.7$. The central diffusive region $0 \le r \le r_{\rm c}$ corresponds to the Earth's inner core for the geodynamo problem and the radiative zone for the solar dynamo problem.

In this simulation, we solve the following (normalized) compressible MHD equations in the outer shell $r_{\rm i} \le r \le 1$ under the rotating frame of reference of the angular velocity $\boldsymbol{\Omega}$, which is in the $+z$ axis.

$$\frac{\partial \rho}{\partial t} = -\nabla \cdot \boldsymbol{f}, \tag{4}$$

$$\frac{\partial \boldsymbol{f}}{\partial t} = -\nabla \cdot (\boldsymbol{v}\boldsymbol{f}) - \nabla p + \boldsymbol{j} \times \boldsymbol{b} + \rho \boldsymbol{g} + 2\boldsymbol{f} \times \boldsymbol{\Omega} + \mu \left[\nabla^2 \boldsymbol{v} + \frac{1}{3} \nabla (\nabla \cdot \boldsymbol{v}) \right], \tag{5}$$

$$\frac{\partial p}{\partial t} = -\boldsymbol{v} \cdot \nabla p - \gamma p \nabla \cdot \boldsymbol{v} + (\gamma - 1) \left[\kappa \nabla^2 T + \eta \boldsymbol{j}^2 + \Phi \right], \tag{6}$$

$$\frac{\partial \boldsymbol{a}}{\partial t} = \boldsymbol{v} \times \boldsymbol{b} - \eta \boldsymbol{j}, \tag{7}$$

with

$$\boldsymbol{g} = -g_0 / r^2 \hat{\boldsymbol{e}}_r, \quad \boldsymbol{b} = \nabla \times \boldsymbol{a}, \quad \boldsymbol{j} = \nabla \times \boldsymbol{b},$$

where ρ (mass density), p (pressure), $\boldsymbol{f}$ (mass flux $\rho \boldsymbol{v}$), and $\boldsymbol{a}$ (vector potential) represent the basic variables. $\hat{\boldsymbol{e}}$ denotes a unit vector in the radial direction. We assume the equation of state of the ideal gas $p = \rho T$ with the specific heat ratio

of $\gamma = 5/3$. The pressure is normalized in such a way that the gas constant is unity in the equation of state. Viscosity μ, electrical resistivity η, and thermal diffusivity κ are constant. Φ denotes the following viscous dissipation function: $\Phi = (\mu/2)(\partial_i v_j + \partial_j v_i)(\partial_i v_j + \partial_j v_i) - (2\mu/3)\,(\partial_k v_k)^2$. This model for the shell region is basically the same as our solar dynamo simulation [25] except that here the magnetic field can diffuse into the central core, $0 \leq r \leq r_{\rm i}$, in which we integrate the diffusion equation for the vector potential:

$$\frac{\partial \boldsymbol{a}}{\partial t} = \eta\, \nabla^2 \boldsymbol{a}, \tag{8}$$

with the same resistivity η as in eq. (7). Other variables are not solved in this core region.

The non-dimensional parameters in the shell region are as follows: Rayleigh number $\mathrm{Ra} = 2 \times 10^4$, Prandtl number $\mathrm{Pr} = 1.0$, Magnetic Prandtl number $\mathrm{Pm} = 5.0$. Angular velocity $\Omega = 0.1$. The initial condition is a hydrostatic equilibrium. The density contrast is $\rho(r_{\rm i})/\rho(1) = 1.19$. We put a weak magnetic "seed" in the shell region, which grows by the MHD dynamo effect. We assume a constant heat flux at $r = r_{\rm i}$ and constant temperature at $r = 1$. For the velocity, the stress-free boundary condition is assumed on $r = r_{\rm i}$ and $r = 1$. The radial magnetic boundary condition is assumed on $r = 1$.

The total grid number is $N_r \times N_\theta \times N_\phi \times \mathrm{Yin/Yang} = 51 \times 84 \times 248 \times 2$ for the Yin–Yang grid, plus $L_x \times L_y \times L_z = 240^3$ for the Zhong grid. The l-cube is divided into $P_x \times P_y \times P_z = 3 \times 3 \times 3$ s-cubes. The program is parallelized with flat MPI: 474 processes for the Yin–Yang grid and 27 processes for the Zhong grid.

The thermal convection motion in the shell region ($r_i \leq r \leq 1$) sets in when the simulation starts. The MHD fluid (the flow and the dynamo-generated magnetic field) reaches a stationary state after a nonlinear saturation. Figure 9(a) shows a meridional cross section of the toroidal component of the magnetic field B_ϕ after the saturation. It is interesting that the toroidal field is formed in the core region ($r \leq r_i$) with an anti-symmetric distribution about the equator: positive in the northern hemisphere and negative in the southern hemisphere. The magnetic field is generated in the convection zone ($r_i \leq r$) and penetrates into the diffusive core region. Although this type of anti-symmetric distribution of the toroidal magnetic field is commonly observed in the convection zone in rapidly rotating spherical shell dynamo systems, it is somewhat surprising in the present case because the rotation rate Ω is relatively small in this test calculation. In fact, the temperature distribution at $r = 0.8$, shown in Fig. 9(b), indicates that the convection motion is not strongly affected by the rotation. This might be a new effect observed when the magnetically diffusive layer below the convection layer is included in the simulation. Details of this simulation will be reported in the future.

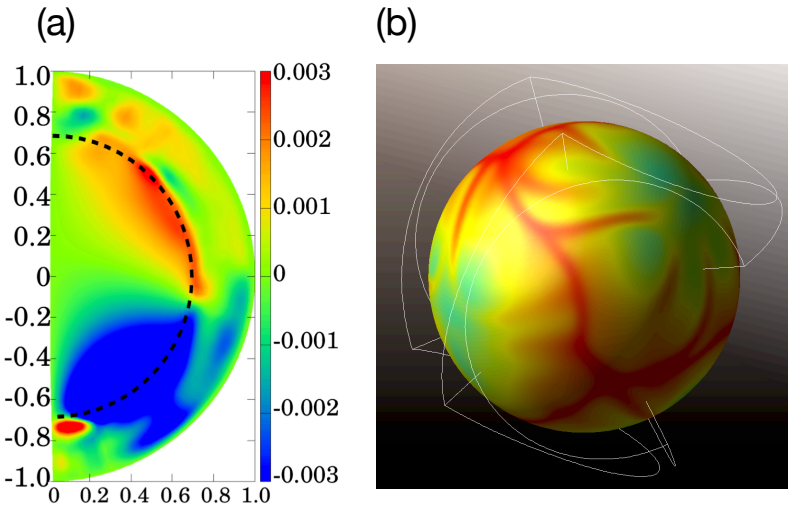

Figure 9: An MHD dynamo simulation with the Yin–Yang–Zhong grid in a rotating sphere. The sphere is composed of two regions that are separated by a sphere of radius $r = 0.7$ (denoted by the dashed line). The outer spherical shell region is a convection zone in which thermal convection leads to a self-generated magnetic field. The inner sphere is a convectively stable (flow-less) region into which the magnetic field can diffuse. (a) Azimuthal component of the dynamo-generated magnetic field B_ϕ. The red and blue colors denote positive and negative B_ϕ, respectively. The penetrated B_ϕ is observed below the core–shell boundary with an anti-symmetric distribution about the equator. (b) Temperature distribution at a sphere of $r = 0.8$, which is the middle of the convection zone.

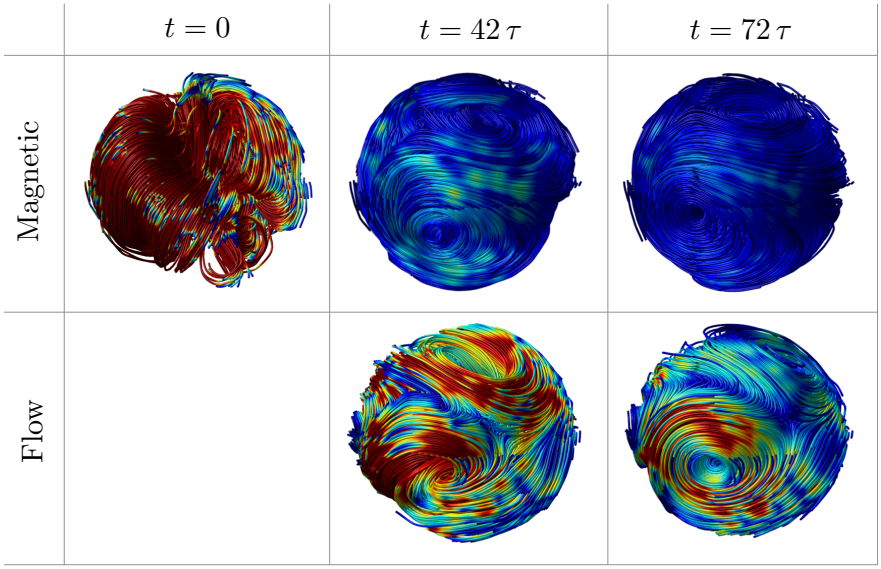

Figure 10: An MHD relaxation simulation inside a sphere. Magnetic field in the initial condition has a non-zero Lorentz force $\boldsymbol{j} \times \boldsymbol{b}$ that drives a flow. The flow and the magnetic field slowly decay because of the dissipations.

4.3 MHD relaxation

The third test that we performed is an MHD relaxation problem inside a sphere with the flow and the magnetic field. In this test, we solve

$$\frac{\partial \rho}{\partial t} = -\nabla \cdot \boldsymbol{f}, \tag{9}$$

$$\frac{\partial \boldsymbol{f}}{\partial t} = -\nabla \cdot (\boldsymbol{v}\boldsymbol{f}) - \nabla p + \boldsymbol{j} \times \boldsymbol{b} + \mu \left[\nabla^2 \boldsymbol{v} + \frac{1}{3} \nabla (\nabla \cdot \boldsymbol{v}) \right], \tag{10}$$

$$\frac{\partial p}{\partial t} = -\boldsymbol{v} \cdot \nabla p - \gamma p \nabla \cdot \boldsymbol{v} + (\gamma - 1) \left[\kappa \nabla^2 T + \eta \boldsymbol{j}^2 + \Phi \right], \tag{11}$$

$$\frac{\partial \boldsymbol{a}}{\partial t} = \boldsymbol{v} \times \boldsymbol{b} - \eta \boldsymbol{j}, \tag{12}$$

in a full sphere $0 \leq r \leq 1$. In the previous test described in § 4.2, only the vector potential $\boldsymbol{a}$ is solved in the core region $0 \leq r \leq r_c$. In contrast, in this test, the full set of the MHD variables, including the mass flux $\boldsymbol{f}$, pressure p and the density ρ, is solved in the core region. The initial condition is a flow-less uniform density ($\rho = 1$) and uniform temperature ($T = 1$) state. To this initial state, a non-zero magnetic field of $\mathcal{O}(10^{-1})$ is imposed, which is constructed from a random superposition of spherical harmonics $Y_\ell^m(\theta, \phi)$ for $(1 \leq \ell \leq 6, \quad 0 \leq |m| \leq 6)$. Since the initial magnetic field has a non-zero Lorentz force, the MHD fluid in the sphere starts flowing at the beginning

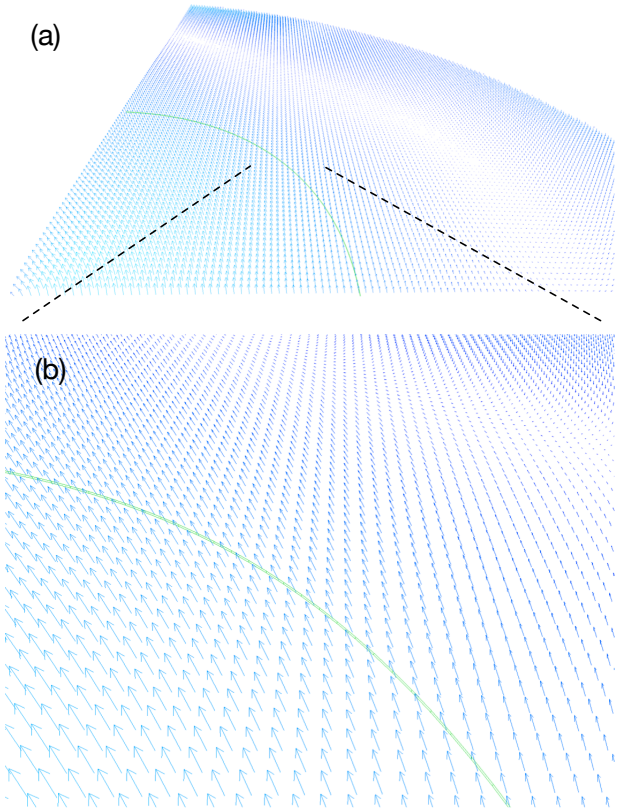

Figure 11: A cross section of magnetic field vectors in the equatorial plane at $t = 42\ \tau$. The green curve denotes the circle of $r = r_c$, i.e., the interface between the Yin-Yang and the Zhong.

of the simulation at time $t = 0$. The total grid number is $N_r \times N_\theta \times N_\phi \times \text{Yin/Yang} + L_x \times L_y \times L_z = 101 \times 104 \times 308 \times 2 + 114^3$. The l-cube is divided into $P_x \times P_y \times P_z = 2 \times 2 \times 2$ s-cubes. The normalized viscosity is $\mu = 9.00 \times 10^{-4}$ and resistivity $\eta = 6.35 \times 10^{-3}$. The total process number used in this test is 320. The radius of the interface between the Yin–Yang and the Zhong is $r_c = 0.35$.

The initial condition with the Lorentz force drives a flow, and then, it slowly decays because of the dissipation. The maximum velocity of the driven flow is $v_{\max} = 0.383$ (Mach number $M = 0.296$) at $t \sim 5\ \tau$, where $\tau = 1/v_{\max}$.

Figure 10 shows snapshots of field lines of the magnetic field $\boldsymbol{b}$ (upper panel) and streamlines of the flow velocity $\boldsymbol{v}$ (lower panel) at $t = 42\ \tau$ and $72\ \tau$ in the relaxation phase. Figure 11 shows a close-up view of the equatorial cross section of the magnetic field $\boldsymbol{b}$ at $t = 42\ \tau$. The green curve denotes the interface between the shell (Yin–Yang) and the core (Zhong).

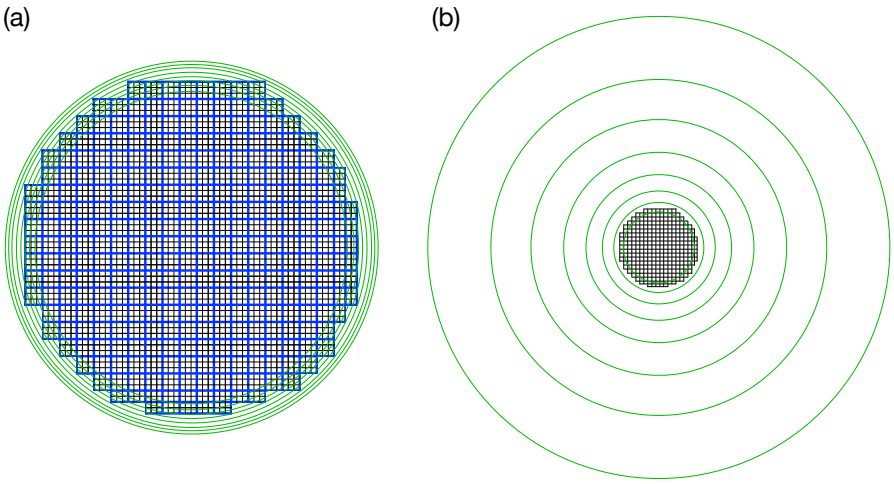

Figure 12: Two extreme cases when the Yin–Yang–Zhong grid is particularly effective. (a) Most of the part inside a sphere is solved by using the Zhong grid, and a thin layer near the spherical boundary is solved by using the Yin–Yang grid. (b) Small-scale dynamics is expected to take place in the center, which is solved by using the Zhong grid, and relatively large-scale structures are expected to form in the outer part, which are solved by using the Yin–Yang grid.

5 Summary and Discussion

For numerical simulations inside a sphere, we have proposed an overset grid system, the Yin–Yang–Zhong grid. The Yin–Yang–Zhong grid is composed of two parts, namely the Yin–Yang grid and the Zhong grid, that are sewn by mutual interpolations.

The Yin–Yang grid is itself an overset grid system that has been successfully applied to various simulations of spherical shell geometry from atmospheres to supernovae. When the Yin-Yang grid is used for a ball geometry including the origin, one has to resolve the severe restriction problem of the time step due to the Courant–Friedrichs–Lewy condition. Therefore, the Yin-Yang grid has a cavity at the center in most applications. In this study, the Zhong component grid is introduced as a patch to cover this cavity. The Zhong component is defined using Cartesian coordinates. Borders of Zhong's grid points are placed just outside the Yin–Yang's cavity with a minimum overlap with the Yin–Yang grid.

The memory for the Zhong grid points is allocated in the unit of cuboid blocks (s-cubes). The s-cubes are in fact decomposed domains for parallel computation, and an MPI process is assigned to each s-cube. The s-cubes that are fully outside the cavity are discarded from the computation at the beginning. Therefore, the Yin–Yang–Zhong grid becomes particularly effective in massively

parallel simulations.

The Yin-Yang-Zhong grid does not guarantee the exact conservation of conserved quantities, unless a conservative interpolation method is adopted in the mutual interpolation between the component grids (Yin, Yang, and Zhong). In this study, we have used the trilinear interpolation for the mutual interpolations in the test calculations in the test calculations. Even with such a simple interpolation scheme without a special care to the local conservation causes no problem in those tests. We have observed no artifact such as an unphysical wave or numerical noise arising at the border between the component grids.

The Yin-Yang-Zhong grid system has a small region where the three component grids overlap. According to the general overset grid methodology [32], special care is not required in such regions. In fact, we have observed no numerical problems in the triple overlap region.

One of the important targets of the Yin–Yang–Zhong grid is an MHD simulation for celestial bodies such as the earth, sun, and other planets and stars because they have a structure similar to that of the Yin–Yang–Zhong: a core at the center and a convection zone at the outer spherical shell.

When one considers applications of the Yin–Yang–Zhong grid to spherical simulations in general, the following question may arise: Do we need the Yin–Yang component above the Zhong component?

There are two extreme cases when the Yin–Yang component plays an essential role. One is the case when a fine resolution is required near the spherical boundary. In the case of fluid dynamics inside a ball, for example, one has to resolve a thin boundary layer on the spherical boundary. The Yin–Yang grid provides a natural grid system that fits the spherical boundary with a stretched grid in the radial direction: See Fig. 12(a). The other case is when a fine-scale phenomenon takes place in the center and, with respect to this scale, relatively large structures are formed in the outer part: see Fig. 12(b). In addition to the MHD simulations for celestial bodies, the Yin–Yang–Zhong grid would provide an effective grid configuration for such cases.

Acknowledgments

The MHD dynamo simulation described in Section 4.2 was performed by T. Furuzono. This work was supported by Grant-in-Aid for Scientific Research (KAKENHI) 23340128.

References

References

[1] S. Goto, N. Ishii, S. Kida, M. Nishioka, Turbulence generator using a precessing sphere, Phys. Fluids 19 (6) (2007) 061705. doi:10.1063/1.2746040.

[2] M. Sturrock, A. Hellander, A. Matzavinos, M. A. J. Chaplain, Spatial stochastic modelling of the Hes1 gene regulatory network: Intrinsic noise can explain heterogeneity in embryonic stem cell differentiation, Journal of The Royal Society Interface 10 (80) (2013) 20120988. doi:10.1098/rsif.2012.0988.

[3] K. Reuter, F. Jenko, A. Tilgner, C. Forest, Wave-driven dynamo action in spherical magnetohydrodynamic systems, Physical Review E 80 (2008). doi:10.1103/PhysRevE.80.056304.

[4] S. F. Hosseinizadeh, A. R. Darzi, F. L. Tan, Numerical investigations of unconstrained melting of nano-enhanced phase change material (NEPCM) inside a spherical container, International Journal of Thermal Sciences 51 (2012) 77–83. doi:10.1016/j.ijthermalsci.2011.08.006.

[5] D. H. Schaubert, D. R. Wilton, A. W. Glisson, A tetrahedral modeling method for electromagnetic scattering by arbitrarily shaped inhomogeneous dielectric bodies, Antennas and Propagation, IEEE Transactions on 32 (1) (1984) 77–85. doi:10.1109/TAP.1984.1143193.

[6] H. Matsui, H. Okuda, MHD dynamo simulation using the GeoFEM platform: Comparison with a spectral method, Pure and Applied Geophysics 161 (11-12) (2004) 2199–2212.

[7] S. Engblom, L. Ferm, A. Hellander, P. Lötstedt, Simulation of stochastic reaction-diffusion processes on unstructured meshes, SIAM Journal on Scientific Computing 31 (3) (2009) 1774–1797.

[8] T. Nakajima, P. H. Roberts, A mapping method for solving dynamo equations, Proceedings of the Royal Society of London A: Mathematical, Physical and Engineering Sciences 448 (1995) 1–28. doi:10.1098/rspa.1995.0001.

[9] D. Komatitsch, J. Tromp, Spectral-element simulations of global seismic wave propagation-I. validation, Geophysical Journal International 149 (2) (2002) 390–412. doi:10.1046/j.1365-246X.2002.01653.x.

[10] A. Fournier, Magnetohydrodynamics in a domain bounded by a spherical surface: A Fourier-spectral element approximation involving a Dirichlet to Neumann operator for the resolution of the exterior problem, Proceedings of European Conference on Computational Fluid Dynamics (2006), pp. 1–20.

[11] P. D. Mininni, D. C. Montgomery, L. Turner, Hydrodynamic and magnetohydrodynamic computations inside a rotating sphere, New Journal of Physics 9 (8) (2007) 303. doi:10.1088/1367-2630/9/8/303.

[12] A. Sakuraba, M. Kono, Effect of the inner core on the numerical solution of the magnetohydrodynamic dynamo, Physics of the Earth and Planetary Interiors 111 (1) (1999) 105–121.

[13] S. Goto, M. Shimizu, G. Kawahara, Turbulent mixing in a precessing sphere, Phys. Fluids 26 (11) (2014) 115106. doi:10.1063/1.4901449.

[14] R. Bayliss, C. Forest, M. Nornberg, E. Spence, P. Terry, Numerical simulations of current generation and dynamo excitation in a mechanically forced turbulent flow, Physical Review E 75 (2) (2007). doi:10.1103/physreve.75.026303.

[15] A. Kageyama, T. Sato, Computer simulation of a magnetohydrodynamic dynamo. II, Physics of Plasmas 2 (1995) 1421.

[16] A. Kageyama, T. Sato, "Yin–Yang grid": An overset grid in spherical geometry, Geochemistry, Geophysics, Geosystems 5 (2004) Q09005, doi:10.1029/2004GC000734.

[17] Y. Baba, K. Takahashi, T. Sugimura, K. Goto, Dynamical core of an atmospheric general circulation model on a Yin–Yang grid, Monthly Weather Review 138 (10) (2010) 3988–4005. doi:10.1175/2010MWR3375.1.

[18] T. Miyagoshi, A. Kageyama, T. Sato, Zonal flow formation in the Earth's core, Nature 463 (7282) (2010) 793–6. doi:10.1038/nature08754.

[19] N. Coltice, T. Rolf, P. J. Tackley, S. Labrosse, Dynamic causes of the relation between area and age of the ocean floor, Science 336 (6079) (2012) 335–8. doi:10.1126/science.1219120.

[20] C. J. Weiss, An overset grid method for global geomagnetic induction, Geophysical Journal International (2014). doi:10.1093/gji/ggu108.

[21] G. J. Golabek, T. Keller, T. V. Gerya, G. Zhu, P. J. Tackley, J. A. Connolly, Origin of the Martian dichotomy and tharsis from a giant impact causing massive magmatism, Icarus 215 (1) (2011) 346–357.

[22] C. C. Reese, C. P. Orth, V. S. Solomatov, Impact megadomes and the origin of the martian crustal dichotomy, Icarus 213 (2) (2011) 433–442.

[23] D. Shiota, R. Kataoka, Y. Miyoshi, T. Hara, C. Tao, K. Masunaga, Y. Futaana, N. Terada, Inner heliosphere MHD modeling system applicable to space weather forecasting for the other planets, Space Weather (2014). http://dx.doi.org/10.1002/2013SW000989 doi:10.1002/2013SW000989.

[24] X. Feng, C. Jiang, C. Xiang, X. Zhao, S. T. Wu, A data-driven model for the global coronal evolution, The Astrophysical Journal 758 (1) (2012) 62.

[25] Y. Masada, K. Yamada, A. Kageyama, Effects of penetrative convection on solar dynamo, The Astrophysical Journal 778 (1) (2013) 11.

[26] H. Hotta, M. Rempel, T. Yokoyama, High-resolution calculations of the solar global convection with the reduced speed of sound technique. I. the structure of the convection and the magnetic field without the rotation, The Astrophysical Journal 786 (1) (2014) 24.

[27] A. Wongwathanarat, N. J. Hammer, and E. Müller, An axis-free overset grid in spherical polar coordinates for simulating 3D self-gravitating flows, Astronomy & Astrophysics 514, A48 (2010). doi:10.1051/0004-6361/200913435.

[28] E. Raymer, Three-dimensional hydrodynamic simulations of accretion in short-period Algols, Monthly Notices of the Royal Astronomical Society 427 (2) (2012) 1702–1712.

[29] D. C. Warren, J. M. Blondin, Three-dimensional numerical investigations of the morphology of type Ia SNRs, Monthly Notices of the Royal Astronomical Society 429 (2013) 3099–3113. doi:10.1093/mnras/sts566.

[30] F. Hanke, B. Müller, A. Wongwathanarat, A. Marek, H.-Th. Janka, SASI activity in three-dimensional neutrino-hydrodynamics simulations of supernova cores, The Astrophysical Journal 770 (1) (2013) 66.

[31] K. Hara, K. Inoue, K. Urahama, Gradient operators for feature extraction from omnidirectional panoramic images, Pattern Recognition Letters 54 (2015) 89–96. doi:10.1016/j.patrec.2014.12.010.

[32] G. Chesshire, W. D. Henshaw, Composite overlapping meshes for the solution of partial differential equations, Journal of Computational Physics 90 (1) (1990) 1–64.

[33] X. Peng, F. Xiao, K. Takahashi, Conservative constraint for a quasi-uniform overset grid on the sphere, Quarterly Journal of the Royal Meteorological Society 132 (616) (2006) 979–996.

[34] D. M. Hall, R. D. Nair, Discontinuous galerkin transport on the spherical Yin–Yang overset mesh, Monthly Weather Review 141 (1) (2013) 264–282.

[35] H. K. Moffatt, Magnetic Field Generation in Electrically Conducting Fluids, Cambridge University Press, Cambridge; New York, 1978.