



A New MHD Model with a Rotated-hybrid Scheme and Solenoidality-preserving Approach

Xueshang Feng^{1,2,3} , Xiaojing Liu^{1,2}, Changqing Xiang¹, HuiChao Li^{1,2}, and Fengsi Wei^{1,3}

¹ SIGMA Weather Group, State Key Laboratory for Space Weather, National Space Science Center, Chinese Academy of Sciences, Beijing 100190, People's Republic of China; fengx@spaceweather.ac.cn

² College of Earth Sciences, University of Chinese Academy of Sciences, Beijing 100049, People's Republic of China

³ HIT Institute of Space Science and Applied Technology, Shenzhen 518055, People's Republic of China

Received 2018 August 12; revised 2018 December 9; accepted 2018 December 21; published 2019 February 4

Abstract

In this paper, the rotated-hybrid scheme is applied for the first time to 3D magnetohydrodynamics (MHD) equations in the finite-volume frame. This scheme is devised by decomposing a cell-face normal vector into two orthogonal directions and combining the Roe solver, a full-wave or complete Riemann solver, and the Rusanov solver, an incomplete Riemann solver, into one rotated-hybrid Riemann solver. To keep the magnetic field divergence-free, we propose two kinds of divergence-cleaning approaches by combining the least-squares reconstruction of magnetic field with the divergence-free constraints. One is the locally solenoidality-preserving method designed to locally maintain the magnetic solenoidality exactly, not just in a least-squares sense, and another is the globally solenoidality-preserving (SP) approach that is implemented by adding a global constraint but abandons the exactness of the locally divergence-free condition. Both SP methods are employed for 3D MHD with a rotated-hybrid scheme in the finite-volume frame. To validate and demonstrate the capabilities of the rotated-hybrid scheme for MHD, we perform an Orszag–Tang MHD vortex problem and a numerical study for the steady-state coronal structures of Carrington rotation 2068 during the solar activity minimum. The numerical tests show the robustness of the proposed scheme and demonstrate the capability of these two SP approaches to keep the magnetic divergence errors to the expected accuracy.

Key words: magnetohydrodynamics (MHD) – Sun: corona – solar wind

1. Introduction

According to Levy et al. (1993), we have learned that the upwind numerical methods for multidimensional computation of systems of hyperbolic conservation laws are categorized into four types: grid-aligned methods, rotation methods, rotation/interpolation methods, and truly multidimensional convection schemes. The grid-aligned method, a common practice in solving multidimensional problems, is based on local 1D Riemann solvers aligned with face normal of the cells, and the upwind direction of this method is normal to the cell face. However, many researchers consider that within this approach 1D solvers lose much of their efficiency, mainly due to the fact that they do not take into account features of the solution propagating transversally to the cell's boundaries. This encourages people to explore other methods. Rather than in a grid-aligned system, the rotation method calculates the flux in a rotated coordinate system dictated by physical flow features that also determine the upwind direction. In the rotation method, the face normal is split into two orthogonal directions: the upwind direction and the direction normal to it. On these two directions, two Riemann problems are solved with the same left and right states. To some extent, the rotation/interpolation approach is similar to the rotation method except for one point: the rotation/interpolation method needs to compute the left and right cell-face states associated with each direction, but the rotation method uses the same left and right states in every direction. Also, there have been many attempts in the literature to build truly multidimensional Riemann solvers (see Balsara 2010, 2012; Balsara et al. 2014; Vides et al. 2015 and the references therein; see also Roe 2017 for a more detailed account on multidimensional upwinding), in

which the interactions at cell corners are taken into account through the approximate solutions of Riemann problems. The truly multidimensional convection scheme can simulate the multidimensional wave propagation as faithfully as possible, but it is usually quite complicated and computationally expensive. Among these four types, the rotation method solves the Riemann problem in a frame rotated about the cell face rather than in a frame normal to the face of a computational cell as done in grid-aligned methods, while this so-called “rotated Riemann solver” is simple and efficient in comparison of the rotation/interpolation method and the truly multidimensional convection schemes. Meanwhile, the rotation method uses flow parameters to determine the upwind direction, which reduces the dependence on grid.

By choosing the velocity difference vector as the upwind direction, Ren (2003) proposed a rotated Roe scheme by applying the Roe solvers along the upwind direction and the other orthogonal to it. Ren (2003) claimed that the shock instabilities can be suppressed by the extra dissipation introduced by the rotation mechanism itself. Nishikawa & Kitamura (2008) suggested a rotated-hybrid solver by using two different Riemann solvers along the two orthogonal directions. In the direction of the velocity difference vector, a dissipative flux solver, such as the Rusanov Riemann solver or the HLL Riemann solver, is directly applied. In the other direction, the Roe Riemann solver is employed with the purpose of preventing the resulting flux from being too dissipative. As pointed out by Nishikawa & Kitamura (2008), the rotated-hybrid solver is free from shock instabilities, and can resolve the boundary layer in fluid flow. Afterward, Huang et al. (2011) proposed a rotated HLLC solver, and Shen et al. (2011) developed a rotated-hybrid low-diffusion ECUSP-HLL

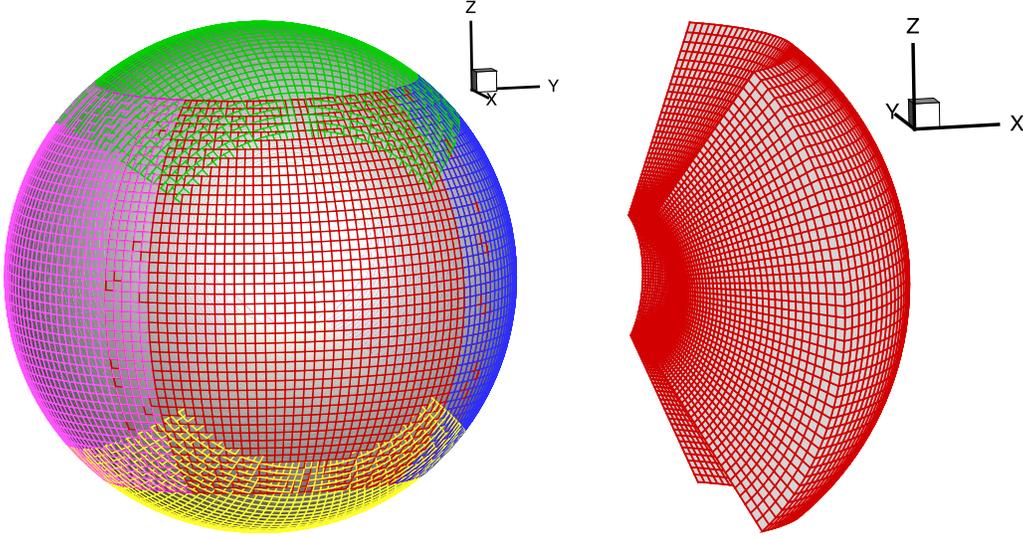


Figure 1. Six-component grid structure with partial overlap (left) and one-component mesh stacked in the r -direction (right).

scheme. In order to reveal the relation between the rotated upwind-differencing direction and the performance of rotated upwind schemes, Zhang et al. (2016) made theoretical and numerical investigations. It should be noted that the rotated Riemann solvers or the rotated-hybrid Riemann solvers have been successfully used in multidimensional hydrodynamic simulations. Here, inspired by the idea of Ren (2003) and Nishikawa & Kitamura (2008), we explore the extension of the rotated-hybrid scheme to the multidimensional MHD equations and further apply it to the steady-state coronal structure simulation.

In the multidimensional MHD simulations, it is well known that maintaining $\nabla \cdot \mathbf{B} = 0$ is a challenge. Failure to do so will cause severe stability problems. In order to keep the magnetic field divergence-free, researchers of MHD simulations have proposed various methods, such as the projection method (Brackbill & Barnes 1980), the constrained transport method (Evans & Hawley 1988), the Powell source term or “eight-wave” method (Powell et al. 1993, 1999), the diffusive method (van der Holst & Keppens 2007), the generalized Lagrange multiplier (GLM) method (Dedner et al. 2002), and the constrained-gradient (CG) method (Hopkins 2016). In the present study, to eliminate $\nabla \cdot \mathbf{B}$ errors, we propose two kinds of divergence-free approaches by combining least-squares reconstruction of the magnetic field with the solenoidal condition.

The paper is structured as follows. Section 2 introduces the MHD governing equations and grid system used in this paper. Numerical implementation is provided in Section 3. Section 4 displays the numerical results of the Orszag–Tang MHD vortex problem and the steady-state corona for CR 2068. Finally, we present conclusions and discussions in Section 5.

2. Governing Equations and Grid System

The symmetrizable form of ideal MHD equations, including the continuity, the momentum, the energy, and the magnetic induction equations, can be written as

$$\frac{\partial \mathbf{U}}{\partial t} + \nabla \cdot \mathbf{F} = \mathbf{S}_{\text{Powell}} + \mathbf{S}_{\text{Others}}, \quad (1)$$

where

$$\mathbf{U} = (\rho, \rho \mathbf{v}, E, \mathbf{B})^T, \\ \mathbf{F} = \begin{pmatrix} \rho \mathbf{v} \\ \rho \mathbf{v} \mathbf{v} + \left(p + \frac{B^2}{2}\right) \mathbf{I} - \mathbf{B} \mathbf{B} \\ \left(E + p + \frac{B^2}{2}\right) \mathbf{v} - (\mathbf{v} \cdot \mathbf{B}) \mathbf{B} \\ \mathbf{v} \mathbf{B} - \mathbf{B} \mathbf{v} \end{pmatrix}.$$

Here ρ , $\mathbf{v} = (v_x, v_y, v_z)$, p , and $\mathbf{B} = (B_x, B_y, B_z)$ are the mass density, velocity, thermal pressure, and magnetic field, respectively. E is the total energy density and is given by $E = \frac{p}{\gamma-1} + \frac{1}{2} \rho \mathbf{v}^2 + \frac{B^2}{2}$. A factor of $1/\sqrt{\mu_0}$ is included in the definition of \mathbf{B} , with $\mu_0 = 4 \times 10^{-7} \pi \text{ H m}^{-1}$ the magnetic permeability of free space. γ is the ratio of specific heats. $\mathbf{S}_{\text{Powell}}$ is the Powell source term (Powell et al. 1999), which reads as follows:

$$\mathbf{S}_{\text{Powell}} = - \begin{pmatrix} 0 \\ \mathbf{B} \\ \mathbf{v} \cdot \mathbf{B} \\ \mathbf{v} \end{pmatrix} \nabla \cdot \mathbf{B}.$$

$\mathbf{S}_{\text{Others}}$ represents the volumetric source arising from the physical considerations.

In the solar coronal numerical study, Equation (1) will be integrated in a spherical shell geometrical computational domain. Following Feng et al. (2010), such spherical shell geometry is partitioned into the six-component grid system, as shown in Figure 1. The six-component grid consists of six identical component meshes with partial overlapping areas. Each component can be treated separately as a low-latitude spherical mesh,

$$\left(\frac{\pi}{4} - \delta \leq \theta \leq \frac{3\pi}{4} + \delta\right) \cap \left(\frac{3\pi}{4} - \delta \leq \phi \leq \frac{5\pi}{4} + \delta\right),$$

where δ is proportionally dependent on the grid spacing entailed for the minimum overlapping area. The six components have the same features, and they can be transformed into

each other by coordinate transformation such that all numerical assignments are identical on each component.

In both the θ - and ϕ -directions, grid points are uniformly distributed: $\theta_i = \theta_{\min} + (i - 1) \Delta\theta$, $i = 1, \dots, N_\theta - 1$, $\phi_i = \phi_{\min} + (i - 1) \Delta\phi$, $i = 1, \dots, N_\phi - 1$, with $\Delta\theta = (\theta_{\max} - \theta_{\min}) / (N_\theta - 2)$, $\Delta\phi = (\phi_{\max} - \phi_{\min}) / (N_\phi - 2)$, where N_θ and N_ϕ are the mesh numbers in latitudinal and longitudinal directions, respectively. In this paper, $\theta_{\min} = \frac{\pi}{4}$, $\theta_{\max} = \frac{3\pi}{4}$, $\phi_{\min} = \frac{3\pi}{4}$, $\phi_{\max} = \frac{5\pi}{4}$, and $N_\theta = N_\phi = 42$. In the r -direction, $r(1) = 1R_s$, $r(i + 1) = r(i) + \Delta r(i)$, where $i = 1, \dots, N_r$. $\Delta r(i) = 0.01R_s$ if $r(i) < 1.1R_s$; $\Delta r(i) = \min(A \times \lg(r(i - 1)), \Delta\theta \times r(i - 1))$ with $A = 0.01/\lg(1.09)$ if $r(i) < 3.5R_s$; $\Delta r(i) = \Delta\theta \times r(i - 1)$ if $r(i) > 3.5R_s$.

In our parallel computations with a distributed memory system, we choose the radial direction as the ‘‘parallel’’ axis. A six-component grid described above is then generated in the remaining 2D space. In this way, it is convenient to implement parallelization in both radial and (θ, ϕ) directions from the construction procedure of the grid system. The small volumes surrounded by the spherical grid points obtained in the spherical coordinates are interpreted as the corresponding control volume elements in the Cartesian coordinates. In this grid system, every grid cell is a hexahedron like a quadrangular frustum pyramid cell.

3. Numerical Implementation

In this section we first introduce the rotated-hybrid scheme in the finite-volume framework. Then, least-squares reconstruction is provided, with particular attention focused on the solenoidality-preserving (SP) approach by combing least-squares reconstruction of the magnetic field with magnetic field divergence-free constraints. Finally, we state the integration of the source term and time-marching method.

3.1. The Rotated-hybrid Scheme

Recall that in the cell-centered finite-volume method, Equation (1) is integrated over a hexahedral computational cell i , surrounded by cells j , yielding

$$\frac{d\bar{U}_i}{dt} = -\frac{1}{V_i} \oint_{\partial\mathcal{V}_i} \mathbf{F} \cdot \mathbf{n} dA + \bar{\mathcal{S}}_{\text{Powell},i} + \bar{\mathcal{S}}_{\text{Others},i}, \quad (2)$$

where $\bar{U}_i = \frac{1}{V_i} \int_{\mathcal{V}_i} \mathbf{U} dV$ is the cell average of \mathbf{U} , V_i is the volume of cell \mathcal{V}_i or cell i for simplicity, $\partial\mathcal{V}_i$ is the boundary of cell i , and $\mathbf{n} = (n_x, n_y, n_z)^T$ is the unit outward vector normal to the boundary $\partial\mathcal{V}_i$. $\bar{\mathcal{S}}_{\text{Powell},i}$ and $\bar{\mathcal{S}}_{\text{Others},i}$ stand for cell-averaged source terms. In Equation (2), the Gauss divergence theorem has been applied to convert the volumetric flux integral to a surface integration over the boundary $\partial\mathcal{V}_i$. Because we are interested in the second-order accurate finite-volume scheme in the present study, it is sufficient to integrate the flux terms in Equation (2) by using the midpoint rule. Equation (2) is now approximated as

$$\frac{d\bar{U}_i}{dt} = -\frac{1}{V_i} \sum_j (\mathbf{F} \cdot \mathbf{n})_{ij} A_{ij} + \bar{\mathcal{S}}_{\text{Powell},i} + \bar{\mathcal{S}}_{\text{Others},i}, \quad (3)$$

where A_{ij} is the area of the interface between cell i and cell j , and $(\mathbf{F} \cdot \mathbf{n})_{ij}$ is the flux across this interface, which is calculated by a rotated-hybrid Riemann solver described below.

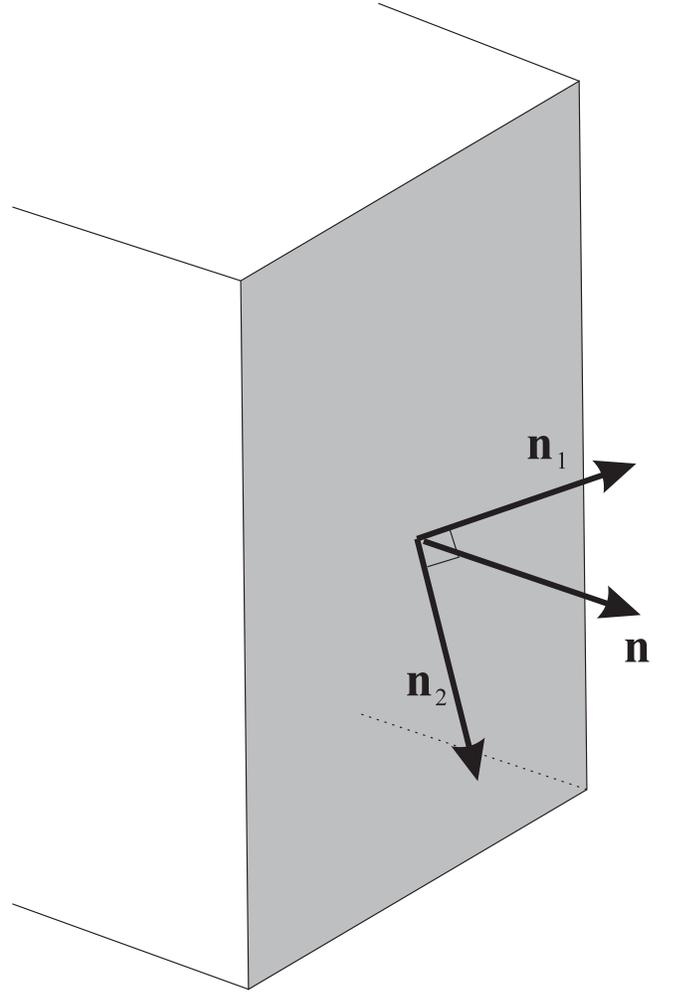


Figure 2. Decomposition of the face normal.

The rotated-hybrid Riemann solver is based on the decomposition of the unit outward normal vector \mathbf{n} into two orthogonal directions, i.e.,

$$\mathbf{n} = \alpha_1 \mathbf{n}_1 + \alpha_2 \mathbf{n}_2, \quad \mathbf{n}_1 \cdot \mathbf{n}_2 = 0,$$

with $|\mathbf{n}_1| = |\mathbf{n}_2| = 1$, $\alpha_1 = \mathbf{n}_1 \cdot \mathbf{n}$, and $\alpha_2 = \mathbf{n}_2 \cdot \mathbf{n}$, as shown in Figure 2. In order to keep the same left and right states in both directions, the vectors \mathbf{n}_1 and \mathbf{n}_2 are chosen to ensure that $\alpha_1 \geq 0$ and $\alpha_2 \geq 0$. According to Ren (2003), the vector \mathbf{n}_1 is defined as follows:

$$\mathbf{n}_1 = \begin{cases} \mathbf{n}, & \text{if } |\Delta\mathbf{v}| \leq \epsilon_1, \\ \Delta\mathbf{v}/|\Delta\mathbf{v}|, & \text{otherwise,} \end{cases}$$

where ϵ_1 is a small positive parameter that is set to 10^{-3} in all the simulations, $\Delta\mathbf{v}$ denotes the velocity difference vector between two adjacent cells, i.e., $\Delta\mathbf{v} = (\Delta v_x, \Delta v_y, \Delta v_z) = (v_{xR} - v_{xL}, v_{yR} - v_{yL}, v_{zR} - v_{zL})$, and $|\Delta\mathbf{v}| = \sqrt{(\Delta v_x)^2 + (\Delta v_y)^2 + (\Delta v_z)^2}$. Another direction \mathbf{n}_2 is calculated, to be defined as perpendicular to \mathbf{n}_1 , by

$$\mathbf{n}_2 = \frac{(\mathbf{n}_1 \times \mathbf{n}) \times \mathbf{n}_1}{|(\mathbf{n}_1 \times \mathbf{n}) \times \mathbf{n}_1|}.$$

Evidently, when $|\Delta v| \leq \epsilon_1$, \mathbf{n}_1 is \mathbf{n} and \mathbf{n}_2 is zero, in which case the method is deduced to the usual flux calculation.

The flux at each interface in Equation (3) is then decomposed correspondingly into the following form:

$$\mathbf{F} \cdot \mathbf{n} = \alpha_1 \mathbf{F} \cdot \mathbf{n}_1 + \alpha_2 \mathbf{F} \cdot \mathbf{n}_2. \quad (4)$$

From Equation (4), it can be seen that we have to solve two Riemann problems, one in \mathbf{n}_1 and the other in \mathbf{n}_2 , to determine $\mathbf{F} \cdot \mathbf{n}_1$ and $\mathbf{F} \cdot \mathbf{n}_2$. According to the rotational invariance of MHD equations (e.g., Tanaka 1994), both of the Riemann problems can be transformed into the 1D problems, namely,

$$\mathbf{F} \cdot \mathbf{n}_1 = \mathbf{T}_1^{-1} \mathbf{f}(\mathbf{T}_1 \mathbf{U}), \quad \mathbf{F} \cdot \mathbf{n}_2 = \mathbf{T}_2^{-1} \mathbf{f}(\mathbf{T}_2 \mathbf{U}),$$

where \mathbf{f} is the flux vector in the x -direction and \mathbf{T}_m is the corresponding rotation matrix associated with \mathbf{n}_m ($m = 1, 2$).

Therefore, Equation (3) is rewritten as

$$\begin{aligned} \frac{d\bar{U}_i}{dt} = & -\frac{1}{V_i} \sum_j \left(\sum_{m=1}^2 \alpha_m \mathbf{T}_m^{-1} \mathbf{f}(\mathbf{T}_m \mathbf{U}) \right)_{ij} \\ & \times A_{ij} + \bar{\mathcal{S}}_{\text{Powell},i} + \bar{\mathcal{S}}_{\text{Others},i}. \end{aligned} \quad (5)$$

If there exists a shock or a shear wave at the cell interface, the shock will propagate in the direction \mathbf{n}_1 and the shear wave will move in the direction \mathbf{n}_2 . Hence, we apply the Rusanov Riemann solver to \mathbf{n}_1 to stabilize the shock and the Roe Riemann solver to \mathbf{n}_2 to promote accuracy. By doing this, we obtain the rotated-hybrid Riemann solver consisting of the Rusanov and Roe Riemann solvers. Seen from Equation (5), the numerical flux is mainly determined by solving a Riemann problem of a specific flux of the x -direction in original MHD equations; thus, for simplicity, we only consider numerical schemes in the x -direction in the following statements.

The Rusanov Riemann solver, devised by Rusanov (1961), is a one-wave approximate Riemann solver. This flux function is very robust for inviscid calculations involving shocks but has an excessive amount of dissipation. Following Yalim et al. (2011), a dissipation-controlled Rusanov flux is expressed as

$$\mathbf{f}(\mathbf{U}_L, \mathbf{U}_R) = \frac{1}{2} [\mathbf{f}(\mathbf{U}_L) + \mathbf{f}(\mathbf{U}_R) - \epsilon_2 |\lambda_{\max}|_{LR} (\mathbf{U}_R - \mathbf{U}_L)], \quad (6)$$

where \mathbf{U}_L and \mathbf{U}_R are the left and right states at interface to be given below. ϵ_2 is a user-defined parameter used for tuning the dissipative part of the Rusanov flux. It varies in the interval (0, 1] and has been set as 0.75 in our code. According to Sitaraman & Raja (2013), $|\lambda_{\max}|$ is the largest local wave speed,

$$|\lambda_{\max}| = |v_x| + \sqrt{\frac{1}{2} \left(a^2 + \frac{|\mathbf{B}|^2}{\rho} + \sqrt{\left(a^2 + \frac{|\mathbf{B}|^2}{\rho} \right)^2 - 4 \frac{B_x^2}{\rho} a^2} \right)},$$

where v_x and B_x are x -components of velocity and magnetic field vectors, $a = \sqrt{\gamma p / \rho}$ is the acoustic wave speed, and $|\mathbf{B}|$ is the magnitude of the magnetic field. $|\lambda_{\max}|_{LR}$ is obtained as the arithmetic averages of values from the left state (L) and the right state (R).

The Roe approximate Riemann solver was first presented by Roe (1981). The Roe solver is a full-wave approximation to the solution of the Riemann problem. It has the characteristic of minimal dissipation so that it can resolve discontinuities better.

The flux function defined in the manner of the Roe scheme is given as follows:

$$\mathbf{f}(\mathbf{U}_L, \mathbf{U}_R) = \frac{1}{2} \left(\mathbf{f}(\mathbf{U}_L) + \mathbf{f}(\mathbf{U}_R) - \sum_{q=1}^8 \mathbf{L}_q (\mathbf{U}_R - \mathbf{U}_L) |\lambda_q| \mathbf{R}_q \right). \quad (7)$$

In Equations (6) and (7), \mathbf{U}_L and \mathbf{U}_R represent the estimates of the conservative variables $\mathbf{U} = (\rho, \rho v_x, \rho v_y, \rho v_z, E, B_x, B_y, B_z)^T$ on the left and right states of the interface. They are obtained using the left and right states, \mathbf{W}_L and \mathbf{W}_R , of the primitive variable $\mathbf{W} = (\rho, v_x, v_y, v_z, p, B_x, B_y, B_z)^T$, which are calculated through the reconstructed procedure shown in Section 3.2 and the Appendix. Similarly, the left and right conservative eigenvectors \mathbf{L}_q and \mathbf{R}_q in Equation (7) are obtained by $\mathbf{L}_q = \mathbf{l}_q \frac{\partial \mathbf{W}}{\partial \mathbf{U}}$, $\mathbf{R}_q = \frac{\partial \mathbf{U}}{\partial \mathbf{W}} \mathbf{r}_q$, $q = 1, 2, \dots, 8$, where \mathbf{l}_q and \mathbf{r}_q are the left and right eigenvectors of Jacobian matrix $\frac{\partial \mathbf{W}}{\partial \mathbf{U}} \frac{\partial \mathbf{f}}{\partial \mathbf{U}} \frac{\partial \mathbf{U}}{\partial \mathbf{W}}$. These eigenvectors \mathbf{l}_q and \mathbf{r}_q are explicitly available (e.g., Roe & Balsara 1996; Powell et al. 1999; Serna 2009). The eigenvectors and eigenvalues are evaluated at an ‘‘interface’’ state that is some combination of the left and right states. Following Powell et al. (1999), we use a simple arithmetic average of left and right states of the primitive variables to compute this interface state. As it is known, Roe’s solver suffers from expansion shocks (Harten 1983). To prevent expansion shocks, an entropy fix is done by replacing $|\lambda_q|$ in Equation (7) with $|\lambda_q^*|$ (for the values of q corresponding to the four magnetoacoustic waves only), where $|\lambda_q^*|$ is given by

$$|\lambda_q^*| = \begin{cases} |\lambda_q|, & |\lambda_q| \geq \frac{\delta \lambda_q}{2}, \\ \frac{\lambda_q^2}{\delta \lambda_q} + \frac{\delta \lambda_q}{4}, & |\lambda_q| < \frac{\delta \lambda_q}{2}, \end{cases}$$

with $\delta \lambda_q = \max(4(\lambda_{qR} - \lambda_{qL}), 0)$.

3.2. The Limited Linear Least-squares Reconstruction

In this subsection, we just simply describe the limited linear reconstruction of the primitive variables of the fluid part. For the reconstruction of the magnetic field, it will be described in the Appendix.

First, the general formula for the linear representation of a primitive solution variable, \mathcal{X} , in a cell, i , is given as follows:

$$\mathcal{X}_i(\mathbf{x}) = \bar{\mathcal{X}}_i + (\nabla \mathcal{X})_i \cdot (\mathbf{x} - \mathbf{x}_i), \quad (8)$$

where $\bar{\mathcal{X}}_i$ is the cell average of primitive variable $\mathcal{X} \in \{\rho, v_x, v_y, v_z, p\}$, \mathbf{x}_i is the cell centroid, and $(\nabla \mathcal{X})_i = \left(\frac{\partial \mathcal{X}}{\partial x}, \frac{\partial \mathcal{X}}{\partial y}, \frac{\partial \mathcal{X}}{\partial z} \right) \Big|_{\mathbf{x}_i}$ is the gradient at the cell centroid, which is determined by a least-squares method (Barth 1993). This reconstruction procedure is carried out with a 27-cell reconstruction stencil that provides better robustness against nonuniformity in the grid and solution gradients that are not aligned with the grid (Ivan et al. 2013). The x -, y -, and z -direction components of $(\nabla \mathcal{X})_i$ are calculated by minimizing the error E_{ji} between the mean value, for cell j in the reconstruction stencil, of the reconstructed polynomial $\mathcal{X}_i(\mathbf{x})$ and the actual volume average $\bar{\mathcal{X}}_j$ as much as possible, and E_{ji}

is expressed as

$$E_{ji} = \bar{\mathcal{X}}_j - \frac{1}{V_j} \int_{\mathcal{V}_j} \mathcal{X}_i(\mathbf{x}) dV, \quad j = 1, 2, \dots, 26.$$

With the notations $\mathbf{x}_i = (x_i, y_i, z_i)$ and $\mathbf{x}_j = (x_j, y_j, z_j)$, we obtain the following overdetermined linear system:

$$\begin{pmatrix} \omega_1(x_1 - x_i) & \omega_1(y_1 - y_i) & \omega_1(z_1 - z_i) \\ \omega_2(x_2 - x_i) & \omega_2(y_2 - y_i) & \omega_2(z_2 - z_i) \\ \vdots & \vdots & \vdots \\ \omega_{N_c}(x_{N_c} - x_i) & \omega_{N_c}(y_{N_c} - y_i) & \omega_{N_c}(z_{N_c} - z_i) \end{pmatrix} \times \begin{pmatrix} \left. \frac{\partial \mathcal{X}}{\partial x} \right|_i \\ \left. \frac{\partial \mathcal{X}}{\partial y} \right|_i \\ \left. \frac{\partial \mathcal{X}}{\partial z} \right|_i \end{pmatrix} = \begin{pmatrix} \omega_1(\bar{\mathcal{X}}_1 - \bar{\mathcal{X}}_i) \\ \omega_2(\bar{\mathcal{X}}_2 - \bar{\mathcal{X}}_i) \\ \vdots \\ \omega_{N_c}(\bar{\mathcal{X}}_{N_c} - \bar{\mathcal{X}}_i) \end{pmatrix}, \quad (9)$$

where $N_c = 26$ is the number of neighbors in the reconstruction stencil and $\omega_j = \frac{1}{|\mathbf{x}_j - \mathbf{x}_i|}$ is the geometric weight that is used to specify the relative importance of the cells in the reconstruction stencil. This system is solved using the least-squares method based on the singular value decomposition (Press et al. 1992). To increase the speed of the calculation, the inverse of the matrix associated with the least-squares problem can be pre-computed and stored since the matrix depends completely on geometry and is the same for all least-squares problems in a given cell i .

Then, to avoid spurious oscillations, the extrapolated value, $\mathcal{X}_i(\mathbf{x})$, obtained by Equation (8) must be constrained with a limiter function, namely,

$$\mathcal{X}_i(\mathbf{x}) = \bar{\mathcal{X}}_i + \phi_i(\nabla \mathcal{X}) \cdot (\mathbf{x} - \mathbf{x}_i). \quad (10)$$

The limiter proposed by Venkatakrishnan (1993) is used in this paper, and the specific procedure is as follows:

1. Find the maximum and minimum cell average values among the cell i and all its neighbors j , i.e., $\bar{\mathcal{X}}_{\max} = \max(\bar{\mathcal{X}}_i, \bar{\mathcal{X}}_j)$, $\bar{\mathcal{X}}_{\min} = \min(\bar{\mathcal{X}}_i, \bar{\mathcal{X}}_j)$.
2. Compute the unlimited reconstructed value at each interface, $\mathcal{X}(x_{ij})$, by using Equation (8), and let $\Delta_- = \mathcal{X}(x_{ij}) - \bar{\mathcal{X}}_i$.
3. Compute the value of ϕ_{ij} at each interface:

$$\phi_{ij} = \begin{cases} \frac{1}{\Delta_-} \left[\frac{(\Delta_+^2 + \epsilon_3^2)\Delta_- + 2\Delta_-^2\Delta_+}{\Delta_+^2 + 2\Delta_-^2 + \Delta_- \Delta_+ + \epsilon_3^2} \right], & \text{with } \Delta_+ = \bar{\mathcal{X}}_{\max} - \bar{\mathcal{X}}_i, \text{ if } \Delta_- > 0, \\ \frac{1}{\Delta_-} \left[\frac{(\Delta_+^2 + \epsilon_3^2)\Delta_- + 2\Delta_-^2\Delta_+}{\Delta_+^2 + 2\Delta_-^2 + \Delta_- \Delta_+ + \epsilon_3^2} \right], & \text{with } \Delta_+ = \bar{\mathcal{X}}_{\min} - \bar{\mathcal{X}}_i, \text{ if } \Delta_- < 0, \\ 1 & \text{if } \Delta_- = 0. \end{cases} \quad (11)$$

$\epsilon_3^2 = (K\Delta h_i)^3$, where Δh_i is the characteristic length of the cell i defined as the diameter of the smallest inscribed sphere of the cell volume, and K is a tunable parameter that is taken as 0.3 in this paper.

4. Select $\phi_i = \min_j(\phi_{ij})$.
5. Obtain the limited reconstruction expression, i.e., Equation (10).

In the actual calculation, Δ_- in formula (11) should be replaced by $\text{sign}(\Delta_-)(|\Delta_-| + \epsilon_4)$ with $\epsilon_4 = 10^{-12}$ to prevent division by a very small value.

In the design of numerical methods for solving the MHD system, it is fundamental to properly handle the magnetic divergence constraint. In particular, in the presence of shocks, numerical methods can produce a large divergence error that may lead to nonphysical flows or negative densities or pressures. Thus, the divergence-free constraint on the magnetic field has to be imposed so as to ensure the accuracy and stability of the numerical schemes at the same time. Here we propose two divergence-cleaning approaches characterized with locally and globally SP properties, called the LSP method and the GSP method, respectively. They are introduced in the Appendix for the clarity of readability.

3.3. Source Terms and Time Integration

In the source terms, $\mathbf{S}_{\text{Others}}$ takes the usual cell averages. As a common practice, the Powell source term, $\bar{\mathbf{S}}_{\text{Powell},i}$, in Equation (2), is computed as follows:

$$\begin{aligned} -\frac{1}{V_i} \int_{\mathcal{V}_i} \mathbf{B}(\nabla \cdot \mathbf{B}) dV &= -\bar{\mathbf{B}}_i \frac{1}{V_i} \sum_j \mathbf{B}_{ij} \cdot \mathbf{n}_{ij} A_{ij}, \\ -\frac{1}{V_i} \int_{\mathcal{V}_i} \mathbf{v} \cdot \mathbf{B}(\nabla \cdot \mathbf{B}) dV &= -(\bar{\mathbf{v}}_i \cdot \bar{\mathbf{B}}_i) \frac{1}{V_i} \sum_j \mathbf{B}_{ij} \cdot \mathbf{n}_{ij} A_{ij}, \\ -\frac{1}{V_i} \int_{\mathcal{V}_i} \mathbf{v}(\nabla \cdot \mathbf{B}) dV &= -\bar{\mathbf{v}}_i \frac{1}{V_i} \sum_j \mathbf{B}_{ij} \cdot \mathbf{n}_{ij} A_{ij}. \end{aligned} \quad (12)$$

In the LSP method, just for the purpose of the test we calculate the Powell source term by substituting the reconstruction expression of \mathbf{B} , obtained by the LSP method, into $\nabla \cdot \mathbf{B}$, i.e.,

$$\begin{aligned} &-\frac{1}{V_i} \int_{\mathcal{V}_i} \mathbf{B}(\nabla \cdot \mathbf{B}) dV \\ &= -\bar{\mathbf{B}}_i \frac{1}{V_i} \int_{\mathcal{V}_i} \left(\frac{\partial B_x}{\partial x} + \frac{\partial B_y}{\partial y} + \frac{\partial B_z}{\partial z} \right) dV, \\ &-\frac{1}{V_i} \int_{\mathcal{V}_i} \mathbf{v} \cdot \mathbf{B}(\nabla \cdot \mathbf{B}) dV \\ &= -(\bar{\mathbf{v}}_i \cdot \bar{\mathbf{B}}_i) \frac{1}{V_i} \int_{\mathcal{V}_i} \left(\frac{\partial B_x}{\partial x} + \frac{\partial B_y}{\partial y} + \frac{\partial B_z}{\partial z} \right) dV, \\ &-\frac{1}{V_i} \int_{\mathcal{V}_i} \mathbf{v}(\nabla \cdot \mathbf{B}) dV \\ &= -\bar{\mathbf{v}}_i \frac{1}{V_i} \int_{\mathcal{V}_i} \left(\frac{\partial B_x}{\partial x} + \frac{\partial B_y}{\partial y} + \frac{\partial B_z}{\partial z} \right) dV, \end{aligned} \quad (13)$$

which does not affect the overall accuracy of the proposed scheme, since the magnetic field is reconstructed with 2-exact polynomial reconstruction.

As for time integration, a second-order Runge–Kutta scheme is employed. We denote the terms on the right-hand side of Equation (5) as $\mathbf{R}_i(\bar{\mathbf{U}})$ and advance by the following scheme:

$$\begin{aligned} \bar{\mathbf{U}}_i^{(1)} &= \bar{\mathbf{U}}_i^n + \Delta t \mathbf{R}_i(\bar{\mathbf{U}}^n), \\ \bar{\mathbf{U}}_i^{n+1} &= \frac{1}{2} \bar{\mathbf{U}}_i^n + \frac{1}{2} (\bar{\mathbf{U}}_i^{(1)} + \Delta t \mathbf{R}_i(\bar{\mathbf{U}}^{(1)})). \end{aligned}$$

Table 1
Four Cases of the MHD Vortex Problem

Cases	Divergence Cleaning	Source Term	MPI Processes	Time Cost (Set the Time Cost of Case 3D-I as 1 Unit)
3D-I	Without	Equation (12)	66	1.0000
3D-II	With LSP	Equation (12)	66	2.1806
3D-III	With LSP	Equation (13)	66	2.1161
3D-IV	With GSP	Equation (12)	66	11.4710

As usual, the time step is defined by the Courant–Friedrichs–Lewy (CFL) stability condition:

$$\Delta t = \text{CFL} \cdot \min_i \frac{\Delta h_i}{\max_{\text{faces}}(|v_n| + c_{f,n})},$$

where $\text{CFL} = 0.5$, Δh_i is the characteristic size of the computational cell, say, taking its value as that in Equation (11), and v_n and $c_{f,n}$ are the normal plasma velocity and the normal fast magnetosonic speed of cell i , respectively.

4. Numerical Results

In this section, we simulate two problems, the Orszag–Tang MHD turbulence and the steady-state solar corona, to validate the rotated-hybrid scheme.

4.1. Orszag–Tang MHD Vortex Problem

The Orszag–Tang vortex system, proposed by Orszag & Tang (1979), is frequently used as a 2D MHD numerical test by many researchers (Jiang & Wu 1999; Tóth 2000; Zhang et al. 2006; Jiang et al. 2010; Zhou & Feng 2014) since it includes many significant features of MHD turbulence. A detailed discussion of the physics of the Orszag–Tang vortex problem goes beyond the subject of the present work. The interested reader may refer to Orszag & Tang (1979). The computational domain is set as $[0, 2\pi] \times [0, 2\pi]$ with periodic boundary conditions in both the x - and y -directions. We use a uniform 400×400 grid. The initial data are as follows:

$$\begin{aligned} \rho(x, y) &= \gamma^2, & v_x(x, y) &= -\sin y, \\ v_y(x, y) &= \sin x, & v_z(x, y) &= 0, \\ p(x, y) &= \gamma, & B_x(x, y) &= -\sin y, \\ B_y(x, y) &= \sin 2x, & B_z(x, y) &= 0, \end{aligned}$$

where $\gamma = 5/3$.

For the Orszag–Tang vortex problem, Jiang & Wu (1999) previously reported numerical instability in their 8×8 eigensystem without the addition of the correction procedure for the magnetic field, and the negative pressure occurred at $t \approx 3.9$ under the grid resolution of 192×192 . In our actual implementation, we also encounter this problem when using the 8×8 Roe scheme to simulate the vortex model, and the negative pressure occurs at $t \approx 4.5$ in our grid resolution. However, the rotated-hybrid scheme without SP (LSP or GSP) can keep running for a long time, which shows the superiority of rotated hybridization. In order to clearly demonstrate the capability of the two divergence-cleaning approaches presented in this paper, we consider several cases listed in Table 1: the rotated-hybrid scheme equipped with or without LSP and GSP, and the source term calculated by Equation (12) or by Equation (13). The computation is completed on the Th-1A

supercomputer from the National Supercomputing Center in TianJin, China, in which each computing node is configured with two Intel Xeon X5670 CPUs (2.93 GHz, six-core). All the cases in the vortex problem utilize 66 cores. For Case 3D-I, it takes about 1.55 hr of wall time to run up to $t = 100$. In order to compare the computational efficiency of these cases, we set the wall time of 3D-I as 1 unit. The last column of Table 1 presents the time cost of all the cases. As can be observed, Case 3D-IV is the most time-consuming. Figure 3 shows the color images of density and thermal pressure obtained at time $t = 3$. Although there is no accepted reference solution for this problem, our results conform to the previous results (Fuchs et al. 2009; Balsara 2010; Jiang et al. 2010; Yang et al. 2017). The solutions evolve in symmetrical patterns. Figure 4 exhibits the density and thermal pressure profiles along the line of $y = 0.625\pi$ at time $t = 3$. In this figure, we can see clearly the shock discontinuities for density and thermal pressure formed near $x = 0.5, 1.6, \text{ and } 4.4$. In Figures 4(a) and (c), we also observe that the shock discontinuities near $x = 0.5$ generated by Cases 3D-II and 3D-III are a little sharper than that attained by Case 3D-I, while the shock discontinuity from Case 3D-IV, as shown in Figures 4(b) and (d), is much sharper than that from Case 3D-I. This is one demonstration that the divergence-free condition for the magnetic field plays an important role in the MHD calculations.

According to Orszag & Tang (1979), the total energy $\int_V E dV$ and the cross-helicity $\int_V \mathbf{v} \cdot \mathbf{B} dV$, where V denotes the whole computational domain, are considered to be conserved globally in ideal MHD. Therefore, as time evolves, how much the conserved variables are retained can be considered as a standard of performance of these cases. Figure 5 displays the temporal evolution of the total energy and the cross-helicity, normalized to the initial ones, for the four cases. Among Cases 3D-I to 3D-IV, only the total energy from Case 3D-I suffers from small-amplitude oscillations before $t = 40$. From Figure 5(a), it is also seen that, using the same LSP method, the temporal evolution of the total energy from Case 3D-III is superior to that from Case 3D-II, and especially, it can be comparable to that from Case 3D-IV. Thus, Equation (13) can also be an alternative for calculating the source term when we adopt the LSP method to constrain the magnetic divergence. The cross-helicity shown in Figure 5(b) has the same change patterns as those exhibited by the total energy. Again, the results shown in Figure 5 demonstrate the effect of keeping the magnetic field divergence-free. Figure 6 shows the temporal evolution of the global kinetic energy, magnetic energy, and thermal energy. It is noticed that the global kinetic, magnetic, and thermal energies all approximate to some constants at later times in the MHD system, which are almost the same as that of Tang & Xu (2000). In order to clearly show the differences between the four cases in the three global energies, Figure 7 presents the time evolution of the

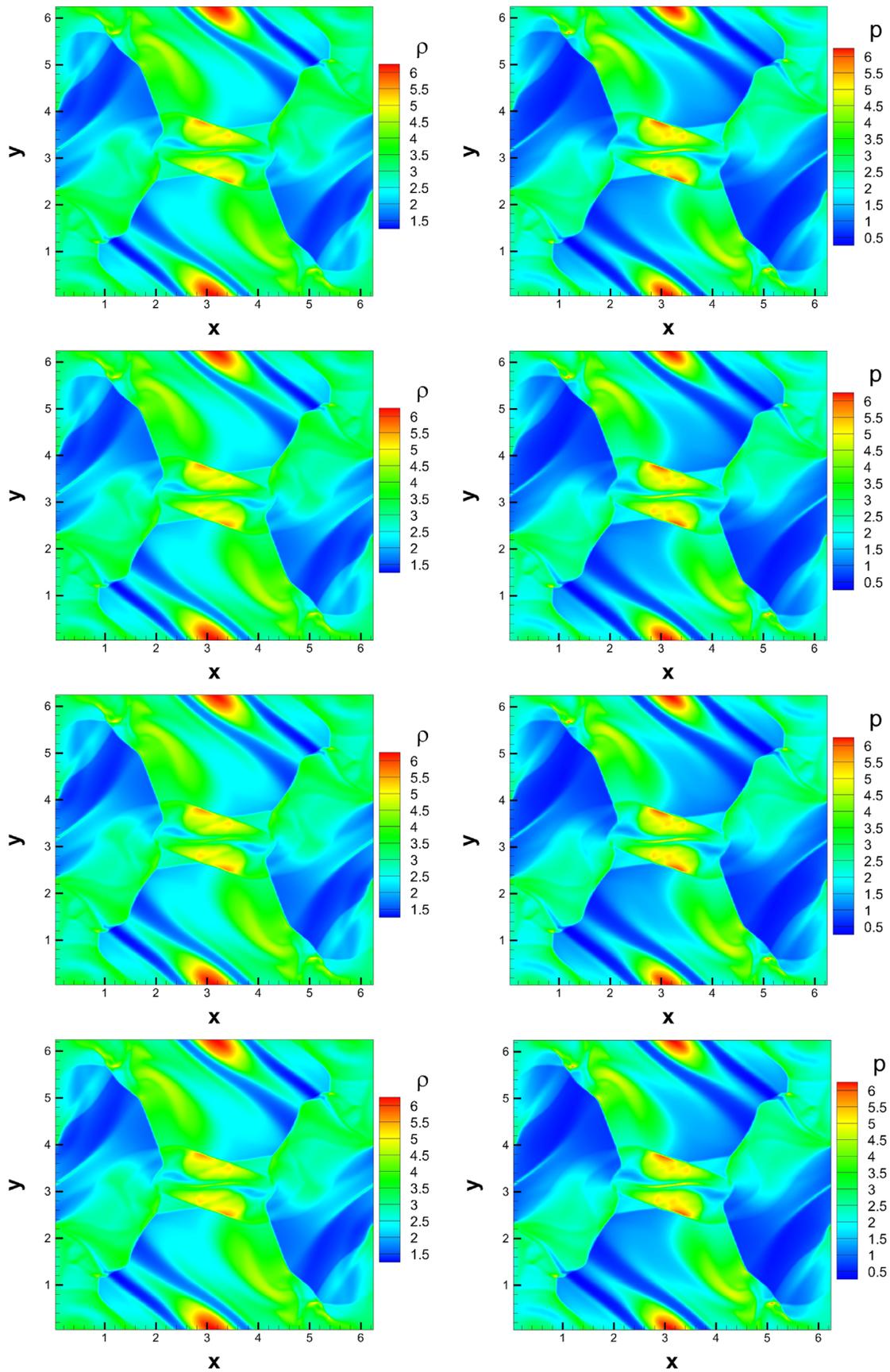


Figure 3. Color images of density (left) and thermal pressure (right) for the Orszag–Tang vortex problem at time $t = 3$ by using the 3D rotated-hybrid scheme with four cases. From top to bottom: 3D-I, 3D-II, 3D-III, and 3D-IV.

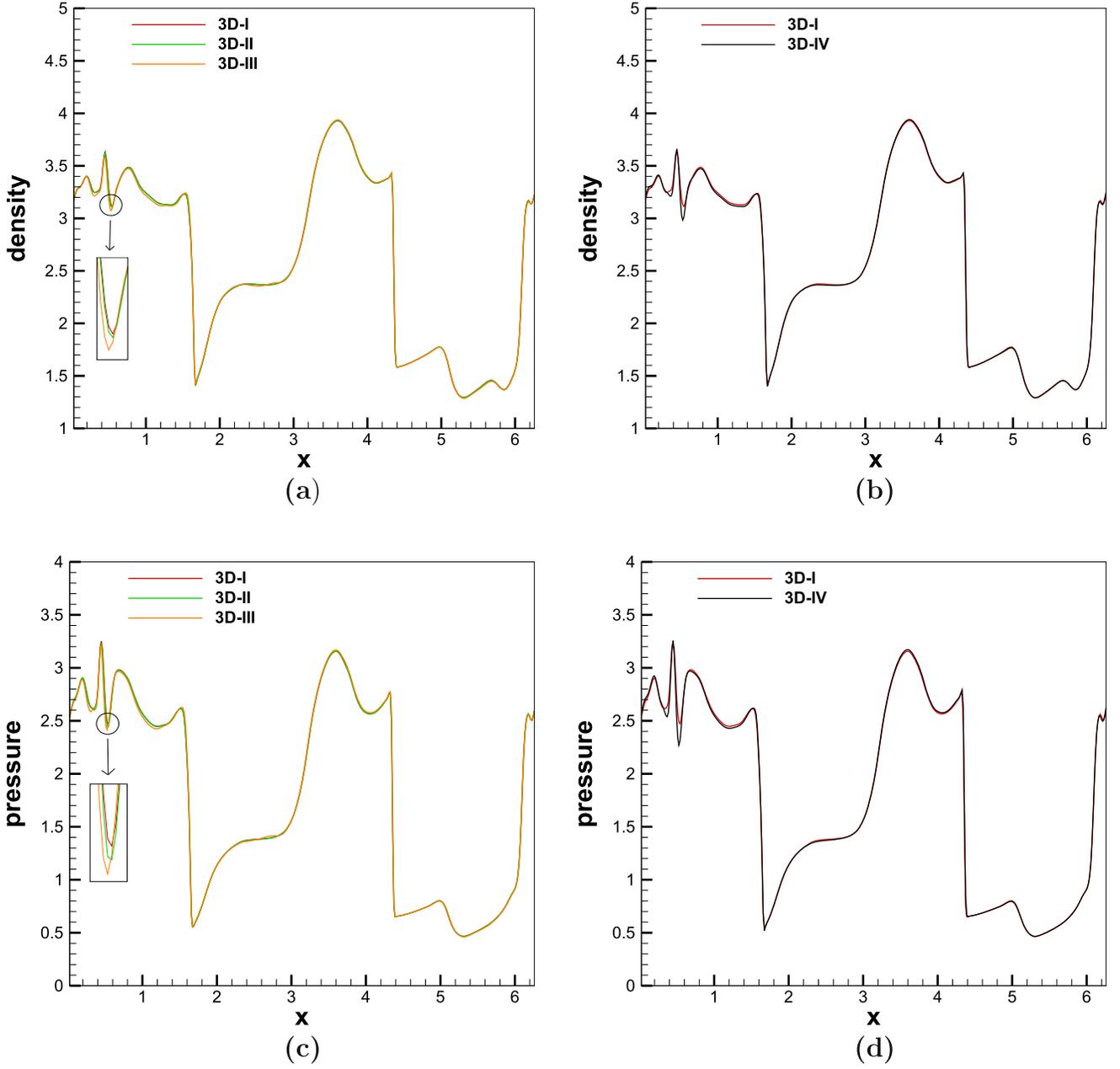


Figure 4. Profiles of density (top) and thermal pressure (bottom) at time $t = 3$ for the Orszag–Tang vortex problem along line $y = 0.625\pi$ by using the 3D rotated-hybrid scheme with Cases 3D-I, 3D-II, 3D-III, and 3D-IV.

relative errors of each global energy between 3D-I, 3D-II, 3D-III, and 3D-IV. Here the modeled results from 3D-IV are chosen as a reference. In Figure 7, the red line denotes the relative error between 3D-I and 3D-IV, which is abbreviated as RE-I-IV. Similarly, the green line denotes the RE-II-IV, and the orange line denotes the RE-III-IV. From the relative errors about the global kinetic energy shown in the left panel of Figure 7, we can observe that RE-III-IV is the smallest; however, the other two relative errors behave similarly but can reach 33%. That is, the global kinetic energy obtained from Case 3D-III is closer to that from Case 3D-IV. The middle panel of Figure 7 shows the relative errors about the global magnetic energy, in which we can see that RE-I-IV, on the whole, is larger than RE-II-IV and RE-III-IV, but they are all lower than 7%. The relative errors about the global thermal energy shown in the right panel of Figure 7 are all below

0.42%, which means that all the cases produce almost the same global thermal energy during the long time evolution. Generally speaking, the temporal evolution of these relative errors reflects the availability of the LSP method (with the high-order reconstruction of the magnetic field) and the source term calculated by Equation (13).

In order to discuss the magnetic divergence, we define the average relative divergence error as

$$\text{Error}(\mathbf{B})^{\text{ave}} = \sum_{i=1}^M \frac{\Delta h_i |\overline{(\nabla \cdot \mathbf{B})}_i|}{|\mathbf{B}_i|} / M, \quad (14)$$

where $\overline{(\nabla \cdot \mathbf{B})}_i$ is given by Equation (20) and M is the total number of cells in the computational domain. Figure 8 shows the temporal evolution of the average relative divergence error, $\text{Error}(\mathbf{B})^{\text{ave}}$, obtained by the four cases. Seen from Figure 8,

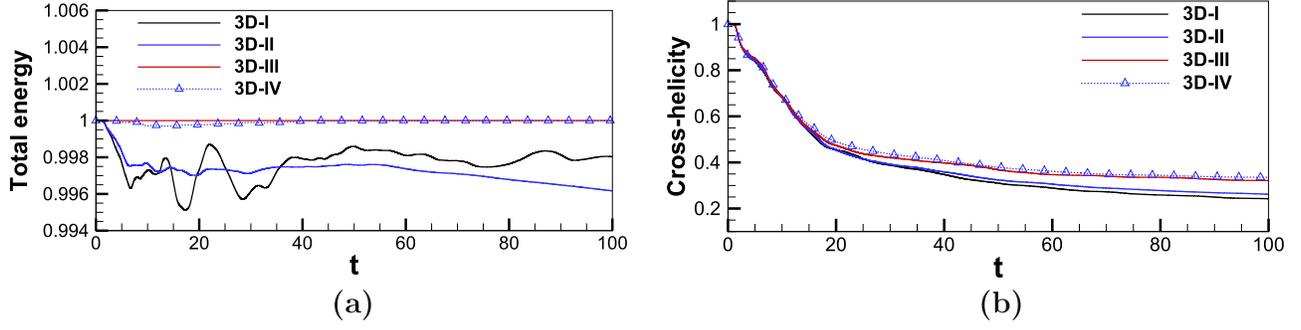


Figure 5. Temporal evolution of (a) the total energy and (b) the cross-helicity, normalized to the initial ones, for the Orszag–Tang vortex problem by using the 3D rotated-hybrid scheme with Cases 3D-I, 3D-II, 3D-III, and 3D-IV.

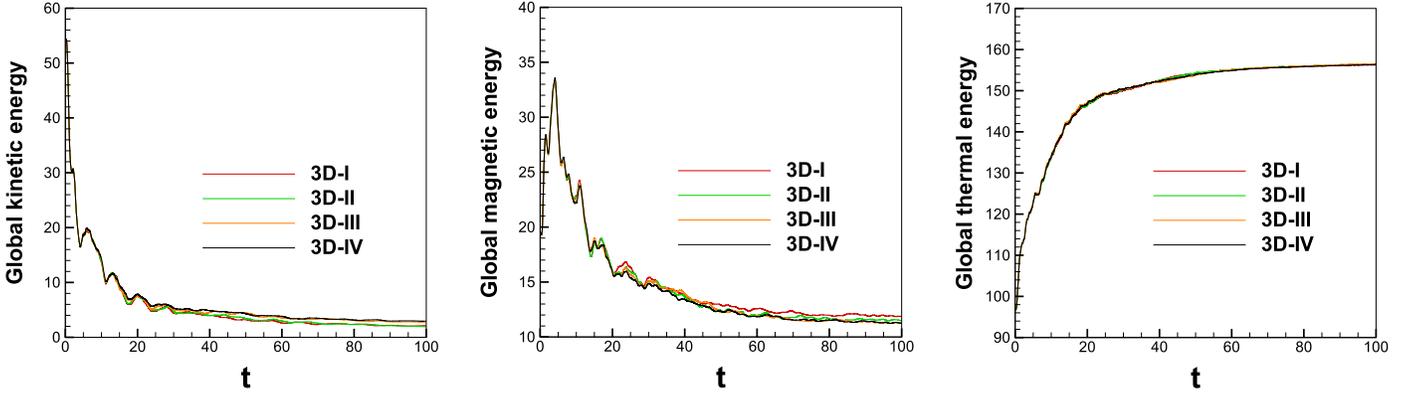


Figure 6. Temporal evolution of global kinetic energy (left), global magnetic energy (middle), and global thermal energy (right) for the Orszag–Tang vortex problem by using the 3D rotated-hybrid scheme with Cases 3D-I, 3D-II, 3D-III, and 3D-IV.

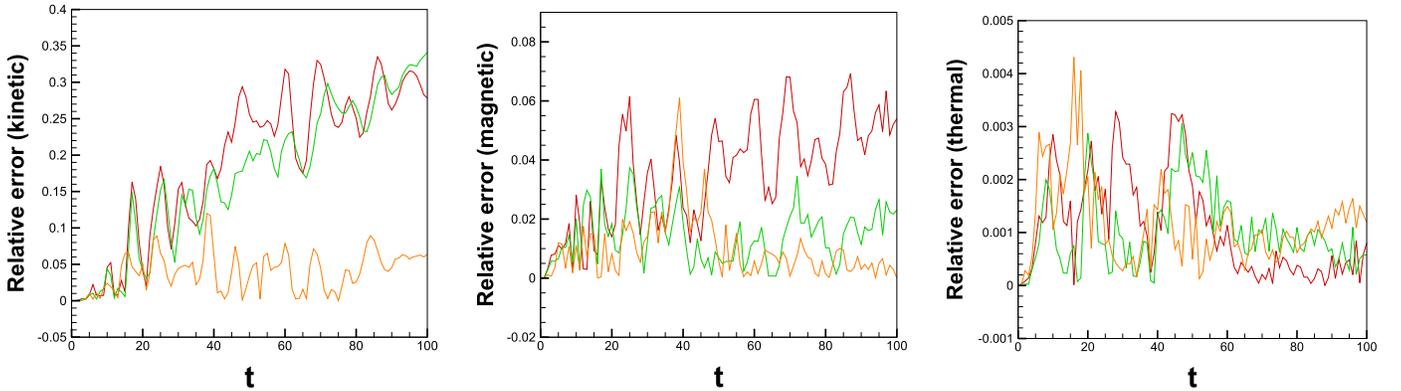


Figure 7. Temporal evolution of the relative errors of global kinetic energy (left), global magnetic energy (middle), and global thermal energy (right) for the Orszag–Tang vortex problem. The red line denotes the relative error between 3D-I and 3D-IV (RE-I-IV). The green line denotes the relative error between 3D-II and 3D-IV (RE-II-IV). The orange line denotes the relative error between 3D-III and 3D-IV (RE-III-IV).

both the LSP method used in Cases 3D-II and 3D-III and the GSP method used in Case 3D-IV have the merits of efficiently reducing the divergence errors. Furthermore, the GSP method is obviously much better than the LSP method.

The Orszag–Tang vortex system is a 2D problem in the space coordinates. To perform the 2D simulation with our 3D code, the z -direction is considered invariant. In the present study, we also carry out the simulation by the 2D code with the same four cases as those in the 3D code, named as 2D-I, 2D-II, 2D-III, and 2D-IV. By using the 2D code, we also achieve almost the same profiles of the corresponding density, thermal pressure, total energy, cross-helicity, global kinetic energy, global magnetic energy, global

thermal energy, and average relative divergence error. The above analysis of numerical results from the 3D code also applies to the modeled results from the 2D code. Through comparison, we observe no significant differences. Hence, the corresponding figures are omitted. Here we only provide the comparison of the modeled density between the 3D and 2D codes. We measure the average relative error of density by $\sum_i \frac{|\bar{\rho}_i^{(2D)} - \bar{\rho}_i^{(3D)}|}{\bar{\rho}_i^{(3D)}}$, where M is the total number of computational cells and $\bar{\rho}_i^{(2D)}$ and $\bar{\rho}_i^{(3D)}$ are the cell-averaged densities obtained from the modeled results of the 2D and 3D codes, respectively. As shown in Table 2, the average relative error of density is around 10^{-4} in magnitude. One thing to note is that the 2D code is much faster than the 3D code,

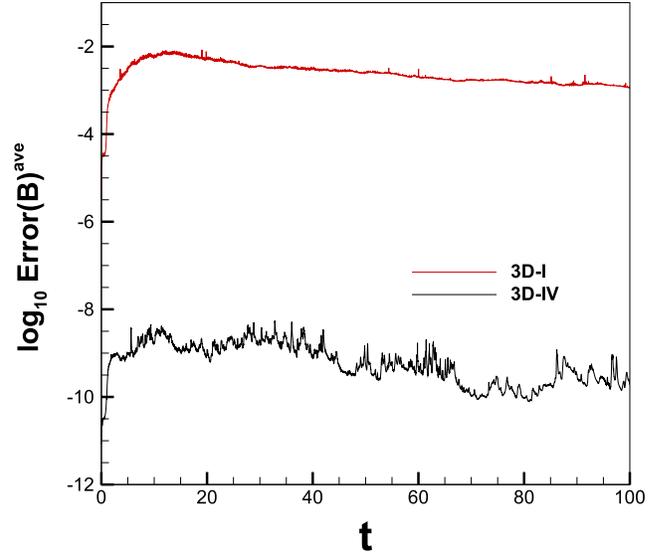
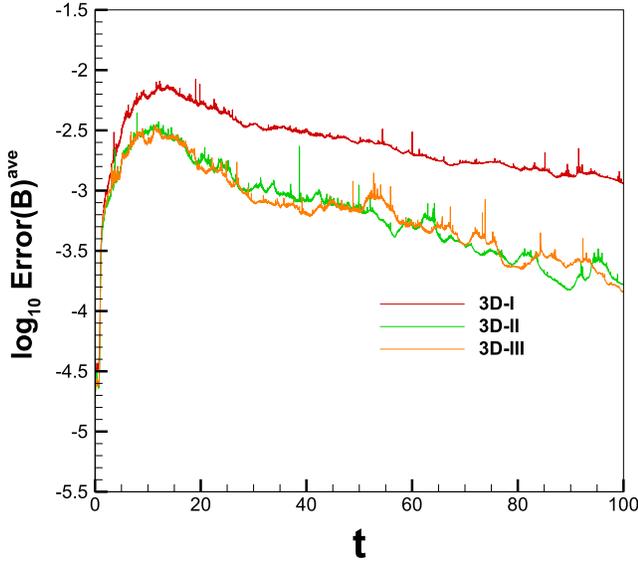


Figure 8. Temporal evolution of the average relative divergence error, $\log_{10}\text{Error}(\mathbf{B})^{\text{ave}}$, for the Orszag–Tang vortex problem by using the 3D rotated-hybrid scheme with Cases 3D-I, 3D-II, 3D-III, and 3D-IV.

Table 2

Average Relative Error of Density and Time Cost between 2D and 3D Codes

3D Code	2D Code	Average Relative Error of Density	Time Cost(t_{2D}/t_{3D})
3D-I	2D-I	1.70×10^{-4}	0.2065
3D-II	2D-II	4.78×10^{-4}	0.1272
3D-III	2D-III	4.76×10^{-4}	0.1341
3D-IV	2D-IV	1.24×10^{-4}	0.2756

which can be seen from the time cost in the last column of Table 2.

4.2. The Steady-state Solar Corona

In this subsection we demonstrate the capability of the rotated-hybrid scheme with the source terms integrated as in Equation (12) to simulate the steady-state corona in CR 2068 during the solar activity minimum. Many authors numerically studied the solar corona and the solar wind in the CR (e.g., Riley et al. 2011, 2012; Pahud et al. 2012). In order to mitigate the occurrence of negative pressure in the numerical study of the solar corona where strong magnetic field dominates, we split the magnetic field \mathbf{B} into the sum of a time-dependent part \mathbf{B}_1 and a time-independent part \mathbf{B}_0 , namely, $\mathbf{B} = \mathbf{B}_1 + \mathbf{B}_0$ (Tanaka 1994; Gombosi et al. 2003; Nakamizo et al. 2009; Feng et al. 2010, 2015). In this situation, $(\rho, \rho\mathbf{v}, E_1, \mathbf{B}_1)^T$ is used for the conservative variables with the corresponding primitive variables $(\rho, \mathbf{v}, p, \mathbf{B}_1)^T$, and $E_1 = \frac{p}{\gamma-1} + \frac{1}{2}\rho\mathbf{v}^2 + \frac{B_1^2}{2}$. In numerical simulation for the solar corona, S_{others} in Equation (1) is given by

$$S_{\text{others}} = \begin{pmatrix} 0 \\ \rho\mathbf{g} - 2\rho\boldsymbol{\Omega} \times \mathbf{v} - \rho\boldsymbol{\Omega} \times (\boldsymbol{\Omega} \times \mathbf{r}) + S_m \\ \rho\mathbf{v} \cdot [\mathbf{g} - \boldsymbol{\Omega} \times (\boldsymbol{\Omega} \times \mathbf{r})] + Q_e + \mathbf{v} \cdot S_m \\ 0 \end{pmatrix}.$$

The variables $\rho, \mathbf{v}, p, \mathbf{B}, \mathbf{r}, t,$ and \mathbf{g} are normalized by the characteristic values $\rho_s, a_s, \rho_s a_s^2, \sqrt{\rho_s a_s^2}, R_s, R_s/a_s,$ and $a_s^2/R_s,$

where $\rho_s, a_s,$ and R_s are the density at the solar surface, sound speed at the solar surface, and solar radius. The solar rotation is considered in this paper with angular velocity $|\boldsymbol{\Omega}| = 2\pi/25.38 \text{ rad day}^{-1}$ and is normalized by $a_s/R_s.$

In order to reflect the magnetic field topology in the heating and acceleration of solar wind (Nakamizo et al. 2009; Feng et al. 2010, 2014, 2017), the momentum-source term S_m and the energy-source term Q_e are given as follows:

$$S_m = M \left(\frac{r}{R_s} - 1 \right) \exp \left(-\frac{r}{L_M} \right) \frac{r}{r},$$

$$Q_e = Q_1 \exp \left(-\frac{r}{L_{Q1}} \right) + Q_2 \left(\frac{r}{R_s} - 1 \right) \exp \left(-\frac{r}{L_{Q2}} \right),$$

where r is the heliocentric distance, $Q_2 = Q_0 C_a,$ $M = M_0 C_a,$ and $C_a = \frac{C'_a}{\max(C'_a)},$ with $C'_a = \frac{(1 - 0.8e^{-(\theta_b/1.0)})^{1.0}}{(1 + f_s)^{2/9}}.$ Here f_s is the

magnetic field expansion factor and reads $f_s = \left(\frac{R_s}{R_{ss}} \right)^2 \frac{B_{R_s}}{B_{R_{ss}}},$ with B_{R_s} and $B_{R_{ss}}$ being the magnetic field strength at the solar surface and at the heliocentric distance $R_{ss} = 2.5R_s,$ respectively. θ_b is the minimum angular separation between an open magnetic field footpoint and its nearest coronal hole boundary. The constant values of $Q_1, Q_0,$ and M_0 are $1.5 \times 10^{-9} \text{ J m}^{-3} \text{ s}^{-1},$ $1.18 \times 10^{-7} \text{ J m}^{-3} \text{ s}^{-1},$ and $7.9 \times 10^{-14} \text{ N m}^{-3},$ respectively. $L_{Q1}, L_{Q2},$ and L_M are all set to be $1R_s.$ It is noted that a similar formula, $C'_a,$ is used to empirically derive the solar wind speed (Tran 2009; Riley et al. 2015).

4.2.1. Initial and Boundary Conditions

The magnetic field, $\mathbf{B}_0,$ in the computational domain is prescribed by using the potential magnetic field solution based on the radial photospheric magnetic field measured by the Global Oscillation Network Group (GONG). The initial condition of \mathbf{B}_1 is set to be 0. For the plasma density $\rho,$ thermal pressure $p,$ and plasma velocity $\mathbf{v},$ we obtain their initial distributions according to Parker's solar wind flow (Parker 1963). The temperature and the number density on the solar surface are $1.3 \times 10^6 \text{ K}$ and $1.5 \times 10^8 \text{ cm}^{-3},$

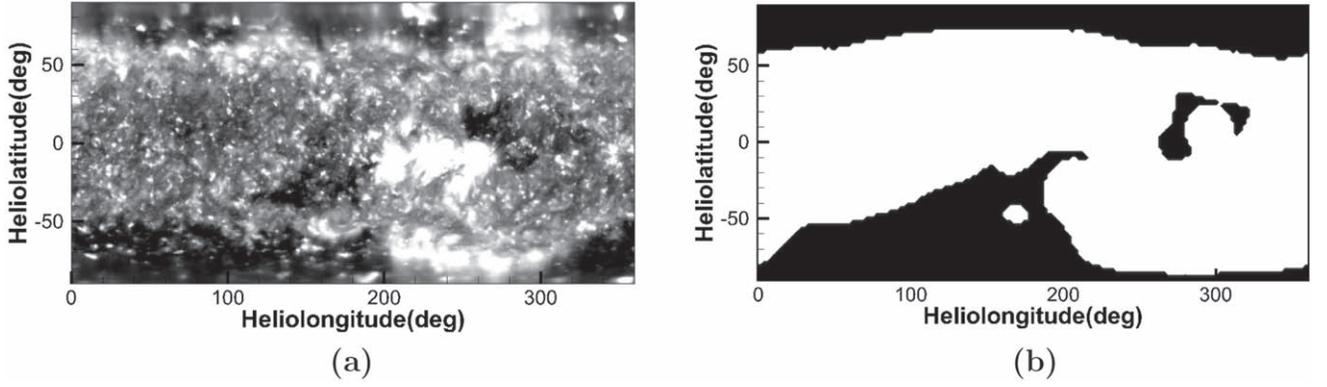


Figure 9. Synoptic maps of the coronal holes at $1R_s$, (a) observed by EUVI/SECCHI on board *STEREO A* and (b) modeled by the MHD model.

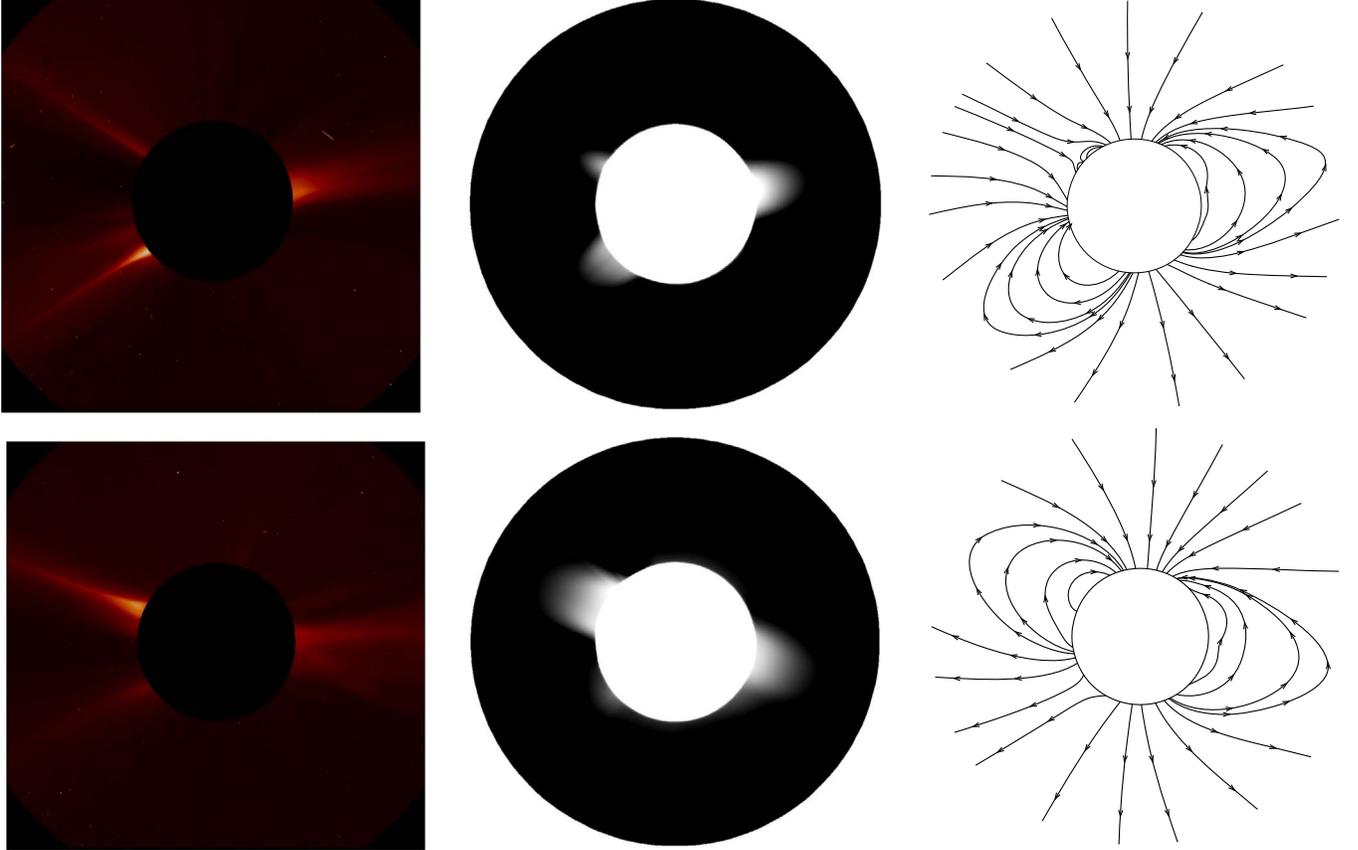


Figure 10. Coronal observations and simulated results on the meridional planes at $\phi = 270^\circ\text{--}90^\circ$ on March 20 (top row) and at $\phi = 180^\circ\text{--}0^\circ$ on March 27 (bottom row). The left and middle columns are the coronal pB images obtained by the observations of LASCO-C2/*SOHO* and the MHD model from $2.3R_s$ to $6R_s$, respectively. The right column displays the simulated magnetic field lines from $1R_s$ to $3R_s$. The black circles in the right column denote the solar surfaces.

respectively. The ratio of specific heats, γ , following Feng et al. (2010), is $\gamma = 1.05$ for $r/R_s \leq 5$, $\gamma = 1.05 + 0.03(r/R_s - 5)$ for $5 < r/R_s \leq 20$, and $\gamma = 1.5$ for $r/R_s > 20$.

For the boundary conditions, the inner boundary at the solar surface is fixed by depending on local flow conditions. That is, for $v_r > 0$,

$$\rho = \rho_s, p = \rho_s a_s^2 / \gamma, \nabla \cdot (\rho \mathbf{v}) = 0, \mathbf{B}_1 = 0,$$

while for $v_r < 0$,

$$\frac{\partial \rho}{\partial r} = 0, \frac{\partial p}{\partial r} = 0, \mathbf{v} = 0, \mathbf{B}_1 = 0,$$

depending on whether the flow along the radial direction is toward or away from the solar surface. It should be mentioned that this kind of boundary setting permits plasma to freely leave the reservoir but allows no “backflow” (e.g., Groth et al. 2000; Feng et al. 2007, 2014).

At the outer boundary, the solar wind parameters are obtained by utilizing the linear extrapolation since the outer boundary is in the supersonic/super-Alfvénic region. In the six-component grid system, there also exist horizontal boundaries at borders in the overlapping parts, and the boundary values of each component grid are reconstructed by using the newly updated values in the neighboring component

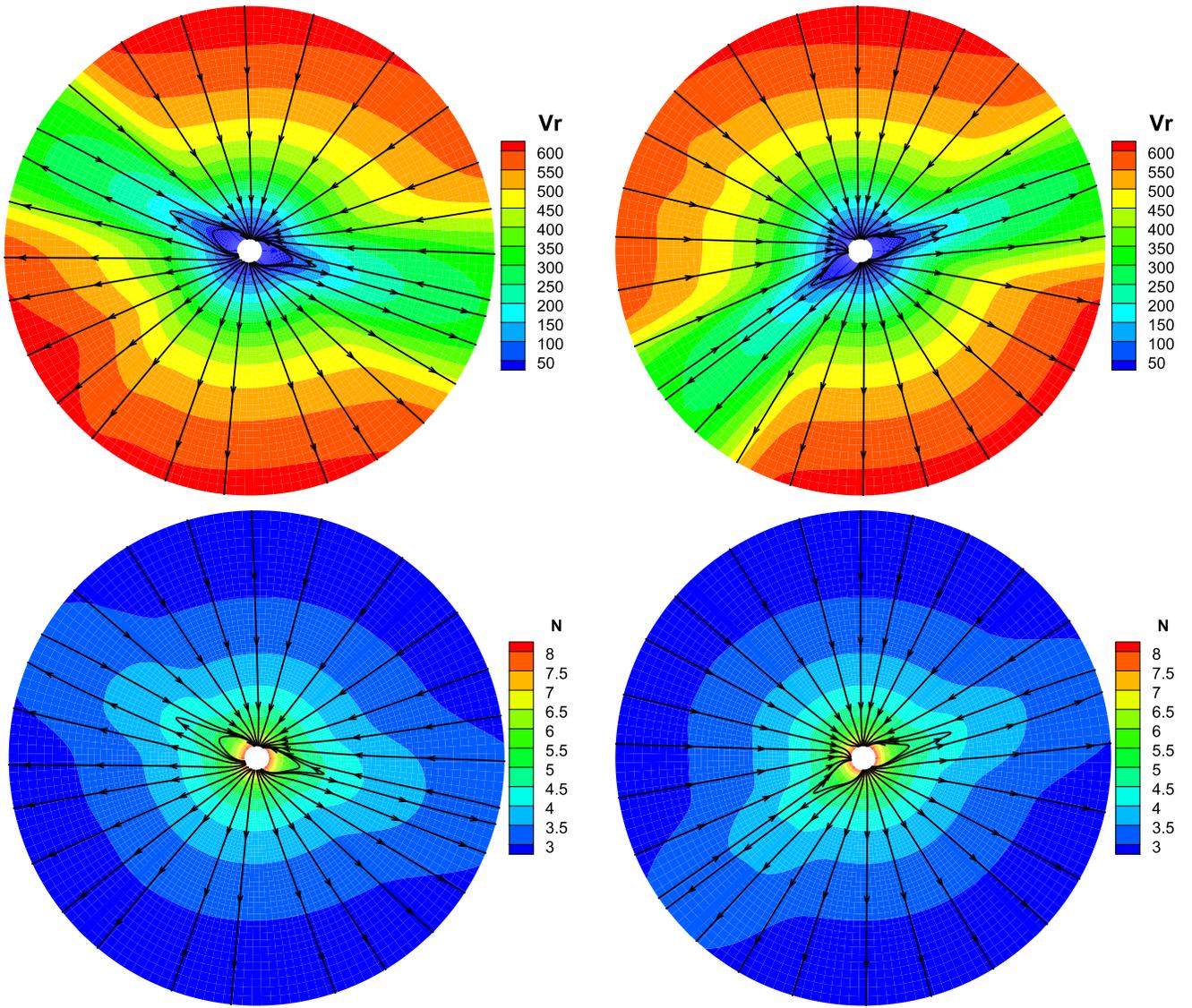


Figure 11. Magnetic field lines, radial speed v_r (km s^{-1}), and number density N (\log_{10}/cm^3) on the two meridional planes of $\phi = 180^\circ\text{--}0^\circ$ (left) and $\phi = 270^\circ\text{--}90^\circ$ (right) from $1R_s$ to $20R_s$.

grids (Feng et al. 2010). Then, we run our code in a time-relaxation manner until a steady-state equilibrium is achieved (e.g., Feng et al. 2007).

In what follows, we employ the rotated-hybrid MHD scheme with the GSP method to investigate the large-scale structures of the solar corona for CR 2068.

4.2.2. The Solar Corona near the Sun

Coronal holes are the dark regions on the solar disk and above the solar limb when measured in extreme-ultraviolet (EUV) and X-ray radiations because coronal holes are often identified as open-field regions at the solar surface (Cranmer 2009; Feng et al. 2017; Lowder et al. 2017). Just as in the previous studies (Luhmann et al. 2002, 2013; Yang et al. 2012; Feng et al. 2015), we use the term ‘‘coronal holes’’ to denote the open-field footpoints of simulated magnetic field lines. Figure 9 presents the synoptic maps of the coronal holes at $1R_s$. Figure 9(a) shows the 195 Å observation from the EUVI instrument on board the *STEREO A* spacecraft. Figure 9(b) shows the simulated coronal

holes, where the open and closed regions are represented by black and white colors, respectively. Seen from the observation and the simulation, the polar coronal holes (PCHs) tend to be asymmetric about both poles. In the longitudinal range of $0^\circ\text{--}220^\circ$, a broad-ridge structure is formed as the southern polar CH extends northward to the equator, and a lobe of the southern polar CH is present at about longitude 210° . Near longitude 270° , we find an isolated low-latitude CH. Comparing the modeled results with observation, the CH configuration is roughly captured by the model. It should be noted that the modeled shapes of both PCHs and the extension of the southern polar CH are similar to those from the model of the potential field source surface (PFSS) in Petrie et al. (2011).

The coronal white-light polarized brightness (pB) images are produced by Thomson scattering from free electrons in the coronal plasma (Linker et al. 1999; Hayes et al. 2001), and these pB images can reflect the distribution of the plasma density qualitatively. The bright regions in the coronal pB images usually correspond to bipolar streamers or pseudo-streamers viewed at the east or west limbs, and the dark regions correspond

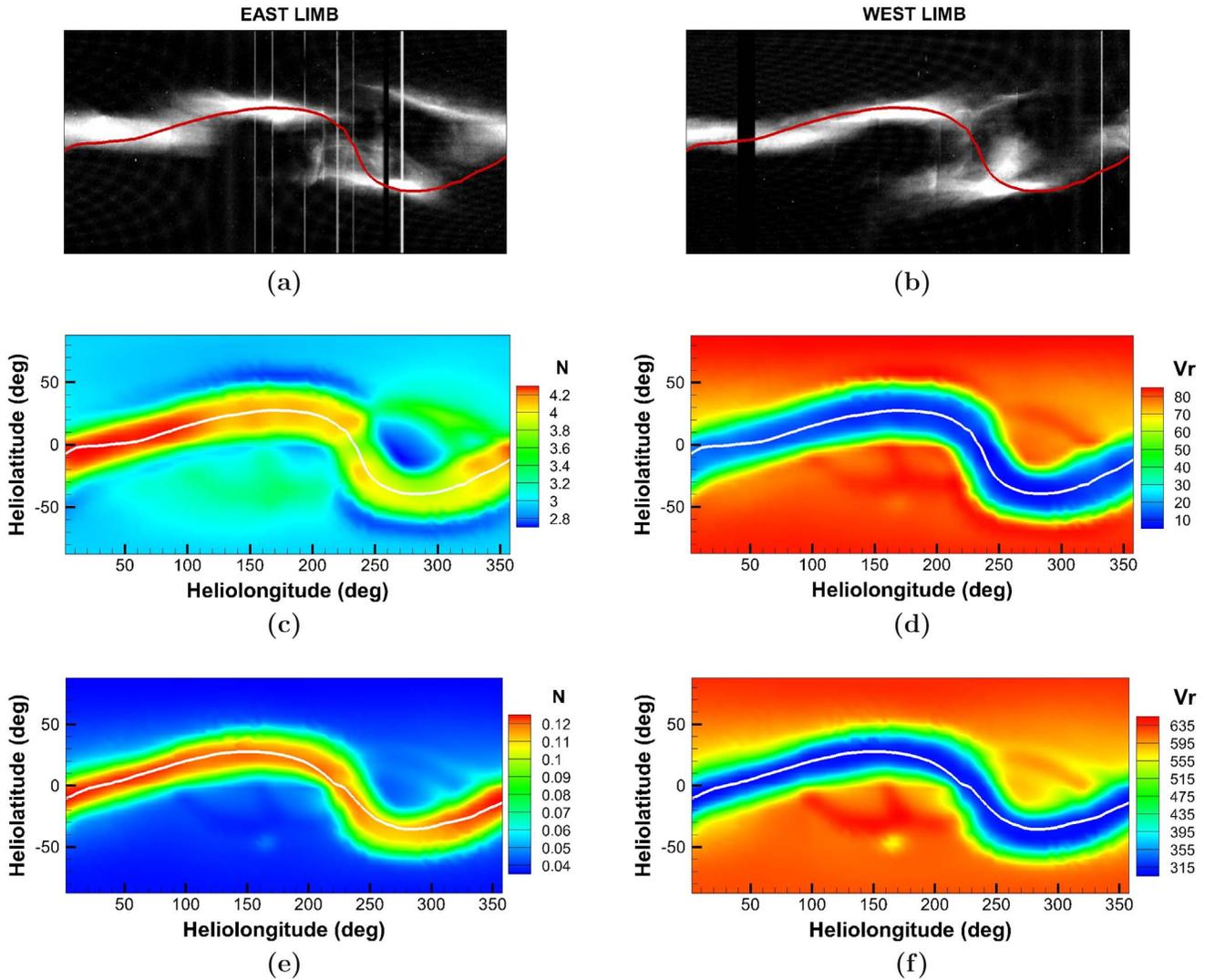


Figure 12. Synoptic maps. (a), (b) White-light pB observations on the surface of $2.6R_s$, at the east and west limbs from *STEREO B* SECCHI COR1; (c) the simulated number density N (unit: 10^5 cm^{-3}) at $2.6R_s$; (d) the radial speed v_r (unit: km s^{-1}) at $2.6R_s$; (e) the simulated number density N (unit: 10^4 cm^{-3}) at $20R_s$; and (f) the radial speed v_r (unit: km s^{-1}) at $20R_s$. The red lines in panels (a) and (b) and the white lines in panels (c)–(f) denote the magnetic neutral lines.

to the low-density open-field regions (Frazin et al. 2007; Feng et al. 2015). Bipolar streamers are coronal structures that separate coronal holes of the opposite magnetic polarities, with a current sheet formed above its cusp, while pseudo-streamers separate coronal holes of the same polarity (Abbo et al. 2015). Figure 10 shows the coronal observations and simulated results on the meridional planes at $\phi = 270^\circ\text{--}90^\circ$ (top row) and $\phi = 180^\circ\text{--}0^\circ$ (bottom row). The first column shows the coronal pB images observed on March 20 (top) and March 27 (bottom) from the LASCO-C2 instrument on board the *SOHO* spacecraft. The second column displays the coronal pB images synthesized by the simulated results of the MHD model from $2.3R_s$ to $6R_s$. The third column presents the simulated magnetic field lines from $1R_s$ to $3R_s$. In the observed and synthesized pB images of the two meridional planes, a wide bright structure extends outward radially at the west limb, along with two separated bright structures at the east limb. The positions of the bright structures are roughly matched in the observed and synthesized images. Comparing with the magnetic topologies in the third column of Figure 10, we infer that the bright structures at the west limb of both meridional planes are formed by streamers. In

the southeast limb of the $\phi = 270^\circ\text{--}90^\circ$ meridional plane and the northeast limb of the $\phi = 180^\circ\text{--}0^\circ$ meridional plane, the bright structures are also associated with streamers. Besides, the bright structure at the northeast limb on the meridional plane of $\phi = 270^\circ\text{--}90^\circ$ results from the high-density pseudo-streamer, and that at the southeast limb on the meridional plane of $\phi = 180^\circ\text{--}0^\circ$ comes from the pseudo-streamer on the plane of $\phi = 200^\circ$ (not shown here). The result from the PFSS model shown in Figure 2 of Petrie et al. (2011) also demonstrated these two pseudo-streamers clearly.

4.2.3. Evolution of the Solar Corona from $2.6R_s$ to $20R_s$

Figure 11 shows the simulation results of radial speed v_r , number density N , and magnetic field lines on two different meridional planes of $\phi = 180^\circ\text{--}0^\circ$ and $\phi = 270^\circ\text{--}90^\circ$ from $1R_s$ to $20R_s$. At the high latitudes, the magnetic field lines open into the interplanetary space, and the high-speed and low-density solar wind pervades this region. On the contrary, the low-speed and high-density solar wind dominates the region near the magnetic neutral line. In Figure 11, helmet streamers stretched by the solar wind can be observed.

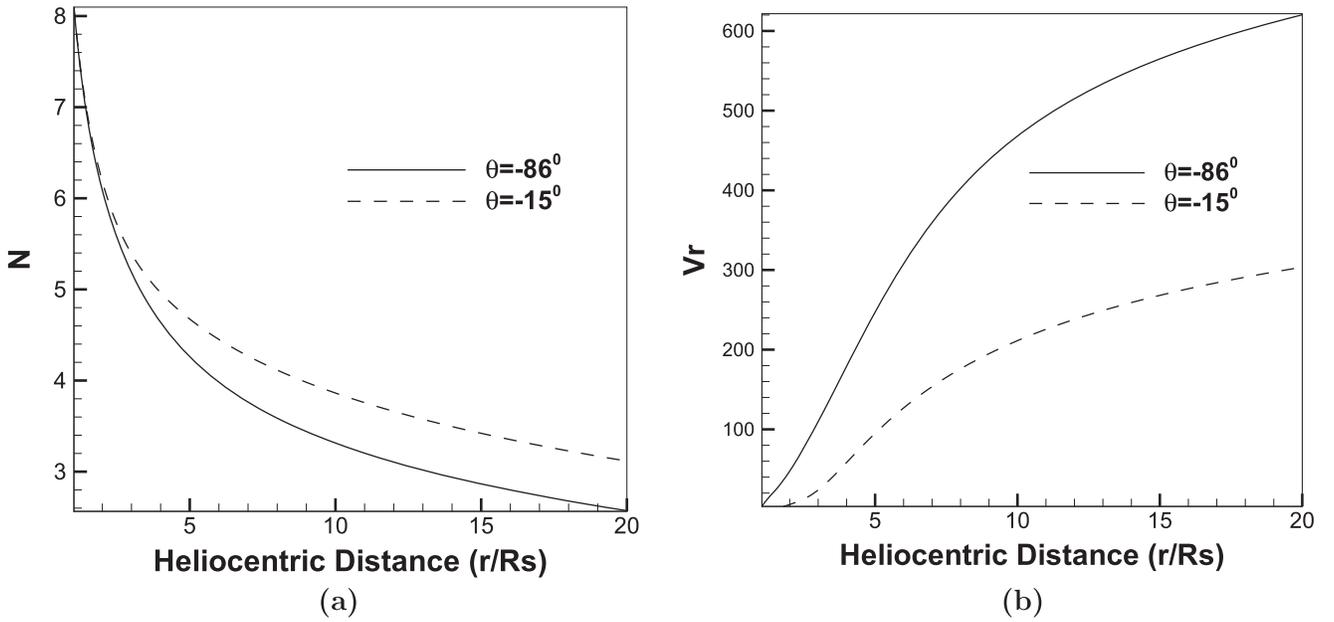


Figure 13. Profiles of the number density N (\log_{10}/cm^3) and radial speed v_r (km s^{-1}) along heliocentric distance with different latitudes $\theta = -86^\circ$ and $\theta = -15^\circ$.

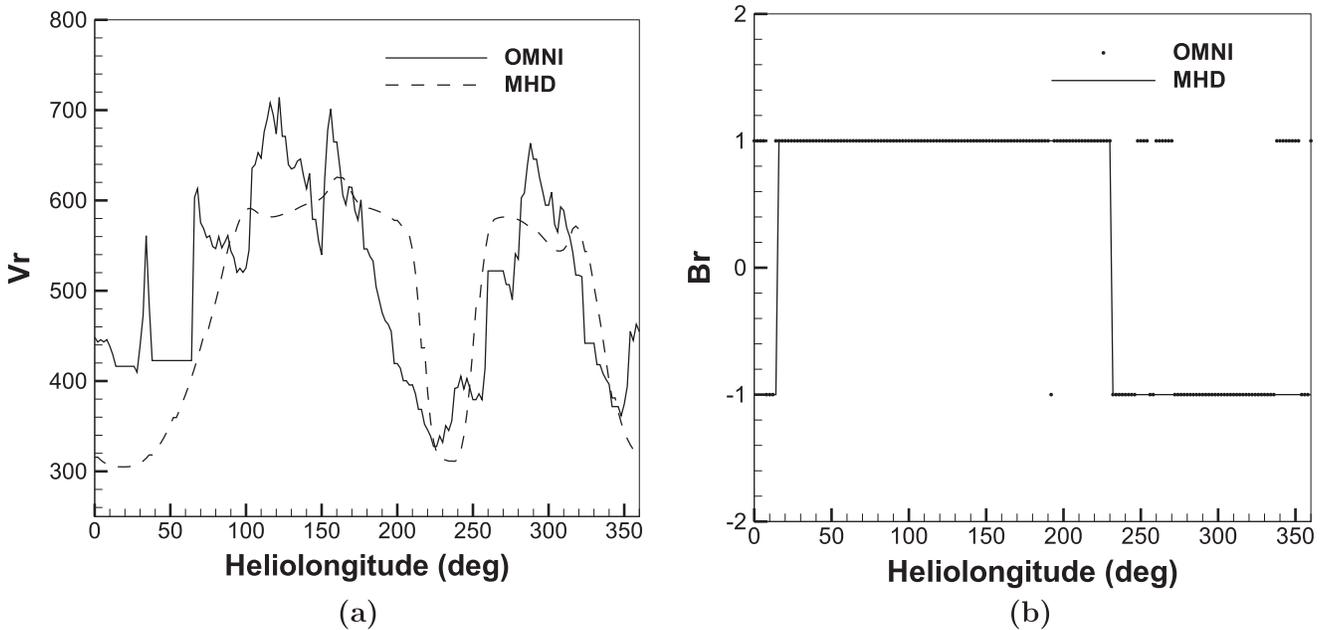


Figure 14. Comparisons between the mapped interplanetary measurements from OMNI and the MHD model's results. (a) Temporal profiles of radial speed v_r (km s^{-1}) from the mapped observations (solid line) and the MHD model (dashed line). (b) Profiles of radial magnetic field polarities obtained from the mapped observations (dotted) and the MHD model (solid line), where “+1” stands for the radial magnetic field away from the Sun and “-1” toward the Sun.

Figure 12 exhibits the synoptic maps of the observation near the corona and MHD numerical solution. Figures 12(a) and (b) show the white-light pB observations on the surface of $2.6R_s$ at the east and west limbs from *STEREO B SECCHI COR1*. The red lines in panels (a) and (b) and the white lines in panels (c)–(f) denote the simulated magnetic neutral lines. It can be clearly seen that the magnetic neutral lines at the east and west limbs are almost surrounded by the bright structures in the white-light pB images. Figures 12(c) and (d) present the number density N and the radial speed v_r on the surfaces of $2.6R_s$. From Figures 12(a)–(d), we can recognize that the locations of the bright structures in the white-light pB images are characterized

by the low velocity and high plasma density in the simulation, and the dark regions are coincident with the locations of the increased flow speed and decreased plasma density. Figures 12(e) and (f) give the number density N and the radial speed v_r at $20R_s$. Comparison between Figures 12(c), (d) and (e), (f) indicates that the number density decreases with the heliocentric distance while radial speed increases. Around the magnetic neutral lines, the high-density and low-speed solar wind flow is located, while at the polar region we can see the low-density and high-speed solar wind flow. In addition, it can be observed that the locations of the peak and the trough of the magnetic neutral line at $2.6R_s$ are $(\theta, \phi) = (28^\circ, 180^\circ)$ and

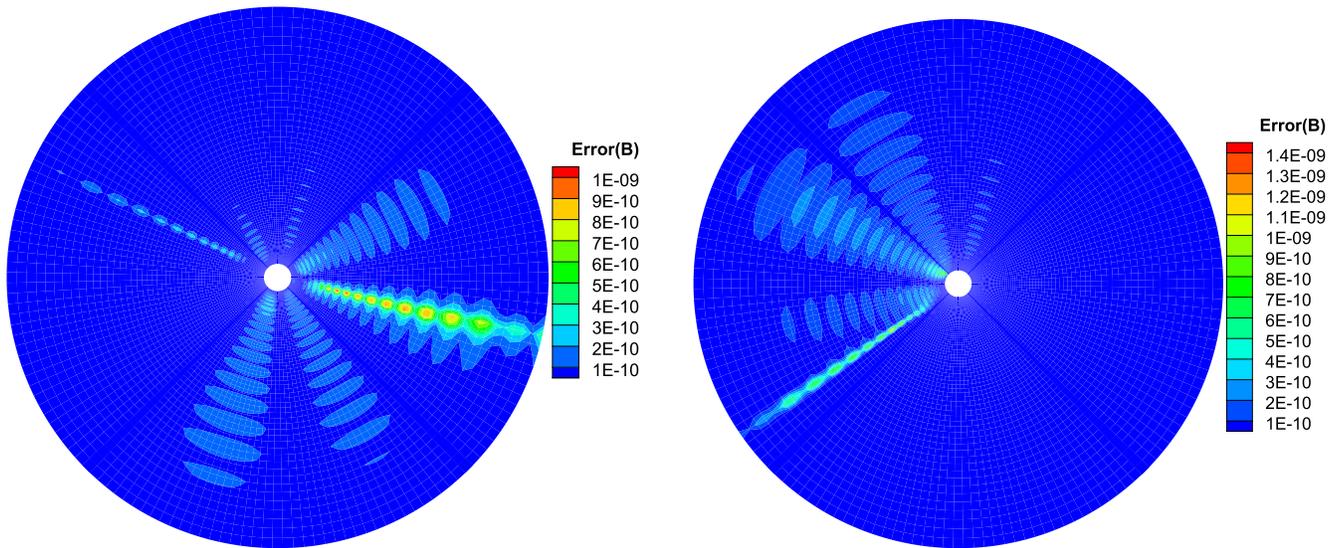


Figure 15. Relative divergence error of the magnetic field, $\frac{\Delta h_i |\nabla \cdot \mathbf{B}_i|}{|\mathbf{B}_i|}$, on the two meridional planes of $\phi = 180^\circ\text{--}0^\circ$ (left) and $\phi = 270^\circ\text{--}90^\circ$ (right) from $1R_s$ to $20R_s$.

$(-40^\circ, 280^\circ)$, while at $20R_s$ they are $(27^\circ, 160^\circ)$ and $(35^\circ, 270^\circ)$. In other words, the magnetic neutral line becomes flatter from $2.6R_s$ to $20R_s$.

Figure 13 displays radial number density and velocity profiles along heliocentric distance at two different latitudes, $\theta = -15^\circ$ (dashed lines) and $\theta = -86^\circ$ (solid lines), at the same longitude $\phi = 1^\circ$, which corresponds to the magnetic neutral line and the open-field region, respectively. From Figure 13(b), the solar wind speed near the magnetic neutral line increases from about 95 km s^{-1} at $5R_s$ to about 304 km s^{-1} at $20R_s$, which basically agrees with the white-light observations with the *LASCO SOHO C2* and *C3* (Porfir'eva et al. 2009). In addition, the solar wind speed in the open-field region rises from about 250 km s^{-1} to about 620 km s^{-1} , which is basically consistent with the previous study on coronal observations (Pätzold et al. 1997).

Since no in situ observation is available in the solar corona during CR 2068, we map the in situ measurements obtained at 1 au back to $20R_s$ by using a ballistic approximation (Yang et al. 2012) to further validate the simulated results. The in situ measurements are publicly available at the OMNI website. Figure 14 shows the temporal profiles of the radial solar wind speed and the radial magnetic field polarities at $20R_s$ obtained from the simulated results and the mapped observation at $20R_s$. In Figure 14(a), the simulated results roughly catch the two high-speed streams and three low-speed streams that occur in the observation. However, an existing defect is that the first low-speed stream obtained by the MHD model is lower than that from the observation. The hit ratio of the simulated radial magnetic field polarities to the observed ones is about 86.2%.

Figure 15 shows the relative divergence error of the magnetic field, $\frac{\Delta h_i |\nabla \cdot \mathbf{B}_i|}{|\mathbf{B}_i|}$, on the two meridional planes of $\phi = 180^\circ\text{--}0^\circ$ (left) and $\phi = 270^\circ\text{--}90^\circ$ (right) from $1R_s$ to $20R_s$. This figure shows that the spatial distribution of the relative divergence error can stay at small magnitude around 10^{-9} . Figure 16 shows the temporal evolution of the average relative divergence error of the magnetic field, which is defined as Equation (14). Once again, it can be seen that the numerical error for the divergence of the magnetic field is kept at the expected accuracy during the time-relaxation process. In the

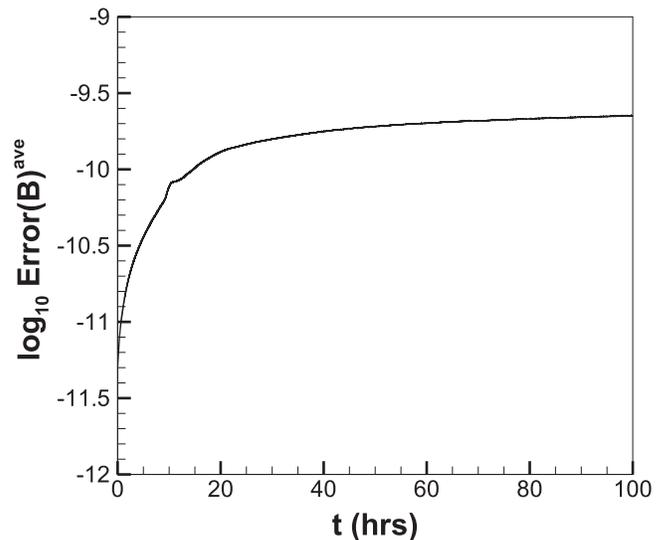


Figure 16. Temporal evolution of the average relative divergence error, $\log_{10} \text{Error}(\mathbf{B})^{\text{ave}}$.

solar coronal simulation, we run our code with the use of 576 MPI processes, and it takes about 50 hr of wall time to obtain a steady-state solution at the physical time 100 hr.

By the way, in the solar coronal simulation, the numerical simulation is carried out for the two divergence-cleaning methods, and the numerical results indicate that the rotated-hybrid schemes with these two approaches are able to provide almost the same large-scale structures morphologically, although only the results obtained from the rotated-hybrid scheme with the GSP method are presented above. However, the time cost and the values of the magnetic divergence errors between them are different. In actual computation, the scheme with the GSP method takes about 5.56 times as much time as that with the LSP method. Nevertheless, the divergence error, $\frac{\Delta h_i |\nabla \cdot \mathbf{B}_i|}{|\mathbf{B}_i|}$, obtained from the GSP method, is smaller than that (around 10^{-3}) from the LSP method since the GSP method constrains the magnetic divergence globally. At the same time, we measure the average relative error of density between them

by $\sum_i \frac{|\bar{\rho}_i^{(\text{LSP})} - \bar{\rho}_i^{(\text{GSP})}| / \bar{\rho}_i^{(\text{GSP})}}{M}$, where M is the total number of computational cells and $\bar{\rho}_i^{(\text{LSP})}$ and $\bar{\rho}_i^{(\text{GSP})}$ are the cell-averaged densities obtained from the modeled results of the LSP method and the GSP method, respectively. It is found that the relative error is about 1.24% at the steady-state solution.

5. Conclusions and Discussions

In this paper, we propose the rotated-hybrid scheme designed for solving MHD equations. In its implementation, the face normal \mathbf{n} in the usual finite-volume frame is decomposed into two orthogonal directions \mathbf{n}_1 and \mathbf{n}_2 . Such decomposition of the face normal is a geometrical procedure so as to keep the same left and right states in both directions, but the physical feature of the flow problem should be taken into account to determine how to choose \mathbf{n}_1 and \mathbf{n}_2 . Thus, to completely define the rotated-hybrid solvers, the direction \mathbf{n}_1 must be determined at every interface by being aligned with some physically meaningful direction, in the present paper, the velocity difference vector (normal to shocks and parallel to shears) taken over two adjacent cells. The rotated-hybrid scheme can be established by utilizing different Riemann solvers along \mathbf{n}_1 and \mathbf{n}_2 . In this way, the rotated Riemann solver approach generates hybrid fluxes in a simple and predigested way like a single Riemann solver and does not require any problem-dependent tuning parameters. If there is a shock or a shear wave at the cell interface, the shock will propagate in the direction of \mathbf{n}_1 and the shear wave will move in the direction of \mathbf{n}_2 . Therefore, when constructing the rotated-hybrid Riemann solver, the more diffusive solver is employed in \mathbf{n}_1 to stabilize the shock, while the less diffusive solver is applied in \mathbf{n}_2 to promote accuracy. In the present paper, we display the description of the rotated-hybrid scheme with a Rusanov solver along \mathbf{n}_1 and a Roe solver along \mathbf{n}_2 . However, our code includes the hybridization of HLL type (along \mathbf{n}_1 direction), such as HLL, HLLC, HLLD, and HLLD, combined with Roe (\mathbf{n}_2 direction), which can be easily implemented in this context.

Another novel point of the present paper is that two SP approaches are established in order to keep the magnetic field divergence-free. Both SP methods combine the least-squares reconstruction of magnetic field with the divergence-free constraints. By fully utilizing $\nabla \cdot \mathbf{B} = 0$, the LSP approach is designed to locally maintain the magnetic solenoidality exactly, not just in a least-squares sense. After $\nabla \cdot \mathbf{B} = 0$ is considered as a built-in property of the current cell of interest, the reduced least-squares system can be calculated much faster at no cost of further increasing the stencil. On the other hand, the GSP method is implemented by adding a global constraint. In such actual computation, the GSP method abandons the exactness of the locally divergence-free condition, and it takes more time than the LSP method. But, the GSP method gives slightly more accurate results for the simulations considered here. Moreover, these two methods are more flexible than the existing divergence-cleaning methods in that they are designed to be exact on any type of grid and also can be extended to high order by fitting high-order polynomials.

Both SP approaches are implemented for 3D MHD with a rotated-hybrid scheme in the finite-volume frame. To validate and demonstrate the capabilities of the rotated-hybrid scheme for MHD, a preliminary study of the Orszag–Tang MHD vortex problem and the steady-state coronal structures of CR 2068 is carried out. In the Orszag–Tang MHD vortex problem,

the rotated-hybrid scheme can keep the Orszag–Tang MHD vortex problem running for a long time without producing negative pressures. Besides, the rotated-hybrid scheme with SP (LSP or GSP) shows a better solution, such as sharper shock discontinuities and conserving the total energy and the cross-helicity better, than that without using SP. For the vortex problem, we also compare the simulation results from the 3D and 2D codes and present the average relative error of density between these two codes. Through comparison, we observe no obvious difference between them. It is worth mentioning that the 2D code is much faster than the 3D code. In the steady-state solar corona simulation, the reasonable comparisons with observations also demonstrate the proposed numerical MHD scheme’s ability of modeling solar corona. More importantly, the numerical tests not only show the robustness of the proposed scheme but also demonstrate the capability of these two SP approaches to keep the magnetic divergence errors to the expected accuracy. Future application of the rotated-hybrid MHD scheme will be devoted to data-driven modeling of solar corona or solar wind (Yang et al. 2012; Feng et al. 2015, 2017; Li & Feng 2018), and particularly magnetic reconnection problems for coronal conditions (Zhang et al. 2011; Cassak & Shay 2012), where controlling numerical dissipation and magnetic field divergence errors are more crucial.

The work is jointly supported by the National Natural Science Foundation of China (grant nos. 41531073, 41731067, 41874202, 41574171, and 41504132) and the Specialized Research Fund for State Key Laboratories. This work utilizes data obtained by the Global Oscillation Network Group (GONG) program, managed by the National Solar Observatory, which is operated by AURA, Inc., under a cooperative agreement with the National Science Foundation. The data were acquired by instruments operated by the Big Bear Solar Observatory, High Altitude Observatory, Learmonth Solar Observatory, Udaipur Solar Observatory, Instituto de Astrofísica de Canarias, and Cerro Tololo Inter-American Observatory. The *STEREO*/SECCHI data are produced by a consortium of the NRL (US), LMSAL (US), NASA/GSFC (US), RAL (UK), UBHAM (UK), MPS (Germany), CSL (Belgium), IOTA (France), and IAS (France). *SOHO* is a project of international cooperation between ESA and NASA. The OMNI data are obtained from the GSFC/SPDF CDAWeb interface at <https://cdaweb.sci.gsfc.nasa.gov/index.html/>. The work was carried out at the National Supercomputer Center in Tianjin, China, and the calculations were performed on TianHe-1 (A). The authors thank the reviewer for constructive suggestions for the improvement of the manuscript.

Appendix The SP Approach

In multidimensional MHD numerical simulations, it is important to maintain the divergence-free condition for the magnetic field. In this appendix, we propose two divergence-free preserving approaches: one is to locally maintain the solenoidality, and another is to try to constrain the divergence-free property globally. The specific process is as follows.

To reconstruct the magnetic field, we use the following Taylor expansion (e.g., Michalak & Ollivier-Gooch 2009;

Ivan et al. 2015):

$$\mathcal{X}_i(\mathbf{x}) = \bar{\mathcal{X}}_i + \sum_{\substack{p_1=0 \\ (0 < p_1 + p_2 + p_3 \leq 2)}}^2 \sum_{\substack{p_2=0 \\ (0 < p_1 + p_2 + p_3 \leq 2)}}^2 \sum_{\substack{p_3=0 \\ (0 < p_1 + p_2 + p_3 \leq 2)}}^2 [(x - x_i)^{p_1} (y - y_i)^{p_2} (z - z_i)^{p_3} - \overline{(x^{p_1} y^{p_2} z^{p_3})}_i] D_{p_1 p_2 p_3}^{\mathcal{X}}, \quad (15)$$

where $\mathcal{X} \in \{B_x, B_y, B_z\}$, $\overline{(x^{p_1} y^{p_2} z^{p_3})}_i = \frac{1}{V_i} \int_{V_i} (x - x_i)^{p_1} (y - y_i)^{p_2} (z - z_i)^{p_3} dV$ are the geometric moments, and $D_{p_1 p_2 p_3}^{\mathcal{X}} = \frac{1}{p_1! p_2! p_3!} \frac{\partial^{p_1 + p_2 + p_3} \mathcal{X}}{\partial x^{p_1} \partial y^{p_2} \partial z^{p_3}} \Big|_{\mathbf{x}_i}$ are the scaled derivatives. Usually, these scaled derivatives are solved by the least-squares technique. Recall that magnetic solenoidality gives us

$$\begin{aligned} \nabla \cdot \mathbf{B} &= \frac{\partial B_x}{\partial x} + \frac{\partial B_y}{\partial y} + \frac{\partial B_z}{\partial z} = 0, \\ \frac{\partial}{\partial x} (\nabla \cdot \mathbf{B}) &= \frac{\partial^2 B_x}{\partial x^2} + \frac{\partial^2 B_y}{\partial x \partial y} + \frac{\partial^2 B_z}{\partial x \partial z} = 0, \\ \frac{\partial}{\partial y} (\nabla \cdot \mathbf{B}) &= \frac{\partial^2 B_x}{\partial x \partial y} + \frac{\partial^2 B_y}{\partial y^2} + \frac{\partial^2 B_z}{\partial y \partial z} = 0, \\ \frac{\partial}{\partial z} (\nabla \cdot \mathbf{B}) &= \frac{\partial^2 B_x}{\partial x \partial z} + \frac{\partial^2 B_y}{\partial y \partial z} + \frac{\partial^2 B_z}{\partial z^2} = 0. \end{aligned} \quad (16)$$

Furthermore, by considering the usual least-squares problem about Equation (15) related to $\mathbf{B} = (B_x, B_y, B_z)$, as well as Equation (16), we obtain the following overdetermined system:

$$\begin{pmatrix} \mathbf{L} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{L} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{L} \\ C_{11} & C_{12} & C_{13} \\ C_{21} & C_{22} & C_{23} \\ C_{31} & C_{32} & C_{33} \\ C_{41} & C_{42} & C_{43} \end{pmatrix}_{(3N_c+4) \times 27} \begin{pmatrix} \mathbf{D}^{B_x} \\ \mathbf{D}^{B_y} \\ \mathbf{D}^{B_z} \end{pmatrix}_{27 \times 1} = \begin{pmatrix} \mathbf{R}^{B_x} \\ \mathbf{R}^{B_y} \\ \mathbf{R}^{B_z} \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix}_{(3N_c+4) \times 1}, \quad (17)$$

where

$$\mathbf{L} = \begin{pmatrix} \omega_1 \overline{(x^1 y^0 z^0)}_{i1} & \omega_1 \overline{(x^0 y^1 z^0)}_{i1} & \cdots & \omega_1 \overline{(x^{p_1} y^{p_2} z^{p_3})}_{i1} & \cdots & \omega_1 \overline{(x^1 y^0 z^1)}_{i1} \\ \omega_2 \overline{(x^1 y^0 z^0)}_{i2} & \omega_2 \overline{(x^0 y^1 z^0)}_{i2} & \cdots & \omega_2 \overline{(x^{p_1} y^{p_2} z^{p_3})}_{i2} & \cdots & \omega_2 \overline{(x^1 y^0 z^1)}_{i2} \\ \vdots & \vdots & \ddots & \vdots & \ddots & \vdots \\ \omega_{N_c} \overline{(x^1 y^0 z^0)}_{iN_c} & \omega_{N_c} \overline{(x^0 y^1 z^0)}_{iN_c} & \cdots & \omega_{N_c} \overline{(x^{p_1} y^{p_2} z^{p_3})}_{iN_c} & \cdots & \omega_{N_c} \overline{(x^1 y^0 z^1)}_{iN_c} \end{pmatrix}_{N_c \times 9}$$

$$\begin{aligned} \overline{(x^{p_1} y^{p_2} z^{p_3})}_{ij} &= \frac{1}{V_j} \int_{V_j} (x - x_i)^{p_1} (y - y_i)^{p_2} (z - z_i)^{p_3} dV - \overline{(x^{p_1} y^{p_2} z^{p_3})}_i, \\ \times j &= 1, 2, \dots, N_c, \end{aligned}$$

$$C_{11} = (1 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0),$$

$$C_{12} = (0 \ 1 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0),$$

$$C_{13} = (0 \ 0 \ 1 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0),$$

$$C_{21} = (0 \ 0 \ 0 \ 2 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0),$$

$$C_{22} = (0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 1 \ 0 \ 0),$$

$$C_{23} = (0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 1),$$

$$C_{31} = (0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 1 \ 0 \ 0),$$

$$C_{32} = (0 \ 0 \ 0 \ 0 \ 2 \ 0 \ 0 \ 0 \ 0 \ 0),$$

$$C_{33} = (0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 1 \ 0),$$

$$C_{41} = (0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 1),$$

$$C_{42} = (0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 1 \ 0),$$

$$C_{43} = (0 \ 0 \ 0 \ 0 \ 0 \ 2 \ 0 \ 0 \ 0 \ 0),$$

$$\mathbf{D}^{B_x} = \left(D_{100}^{B_x} \ D_{010}^{B_x} \ D_{001}^{B_x} \ D_{200}^{B_x} \ D_{020}^{B_x} \ D_{002}^{B_x} \ D_{110}^{B_x} \ D_{011}^{B_x} \ D_{101}^{B_x} \right)^T,$$

$$\mathbf{D}^{B_y} = \left(D_{100}^{B_y} \ D_{010}^{B_y} \ D_{001}^{B_y} \ D_{200}^{B_y} \ D_{020}^{B_y} \ D_{002}^{B_y} \ D_{110}^{B_y} \ D_{011}^{B_y} \ D_{101}^{B_y} \right)^T,$$

$$\mathbf{D}^{B_z} = \left(D_{100}^{B_z} \ D_{010}^{B_z} \ D_{001}^{B_z} \ D_{200}^{B_z} \ D_{020}^{B_z} \ D_{002}^{B_z} \ D_{110}^{B_z} \ D_{011}^{B_z} \ D_{101}^{B_z} \right)^T,$$

$$\mathbf{R}^{B_x} = \begin{pmatrix} \omega_1 (\overline{B_{x1}} - \overline{B_{xi}}) \\ \omega_2 (\overline{B_{x2}} - \overline{B_{xi}}) \\ \vdots \\ \omega_{N_c} (\overline{B_{xN_c}} - \overline{B_{xi}}) \end{pmatrix}, \quad \mathbf{R}^{B_y} = \begin{pmatrix} \omega_1 (\overline{B_{y1}} - \overline{B_{yi}}) \\ \omega_2 (\overline{B_{y2}} - \overline{B_{yi}}) \\ \vdots \\ \omega_{N_c} (\overline{B_{yN_c}} - \overline{B_{yi}}) \end{pmatrix},$$

$$\mathbf{R}^{B_z} = \begin{pmatrix} \omega_1 (\overline{B_{z1}} - \overline{B_{zi}}) \\ \omega_2 (\overline{B_{z2}} - \overline{B_{zi}}) \\ \vdots \\ \omega_{N_c} (\overline{B_{zN_c}} - \overline{B_{zi}}) \end{pmatrix}.$$

Obviously, the last four rows in Equation (17) come from Equation (16). Because Equation (16) must be satisfied exactly, not just in a least-squares sense, we apply the elimination method to the overdetermined system (17) such that some scaled derivatives can cancel out. For example, we utilize $\frac{\partial B_x}{\partial x}$, $\frac{\partial^2 B_x}{\partial x^2}$, $\frac{\partial^2 B_x}{\partial x \partial y}$, $\frac{\partial^2 B_x}{\partial x \partial z}$ for such elimination. Then, we

can obtain the following reduced overdetermined system:

$$\begin{pmatrix} \tilde{\mathbf{L}}_1 & \tilde{\mathbf{L}}_2 & \tilde{\mathbf{L}}_3 \\ \mathbf{0} & \mathbf{L} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{L} \end{pmatrix}_{3N_c \times 23} \begin{pmatrix} \tilde{\mathbf{D}}^{B_x} \\ \mathbf{D}^{B_y} \\ \mathbf{D}^{B_z} \end{pmatrix}_{23 \times 1} = \begin{pmatrix} \mathbf{R}^{B_x} \\ \mathbf{R}^{B_y} \\ \mathbf{R}^{B_z} \end{pmatrix}_{3N_c \times 1}, \quad (18)$$

where

$$\begin{aligned}
\tilde{\mathbf{L}}_1 &= \begin{pmatrix} \omega_1(\widehat{x^0y^1z^0})_{i1} & \omega_1(\widehat{x^0y^0z^1})_{i1} & \omega_1(\widehat{x^0y^2z^0})_{i1} & \omega_1(\widehat{x^0y^0z^2})_{i1} & \omega_1(\widehat{x^0y^1z^1})_{i1} \\ \omega_2(\widehat{x^0y^1z^0})_{i2} & \omega_2(\widehat{x^0y^0z^1})_{i2} & \omega_2(\widehat{x^0y^2z^0})_{i2} & \omega_2(\widehat{x^0y^0z^2})_{i2} & \omega_2(\widehat{x^0y^1z^1})_{i2} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ \omega_{N_c}(\widehat{x^0y^1z^0})_{iN_c} & \omega_{N_c}(\widehat{x^0y^0z^1})_{iN_c} & \omega_{N_c}(\widehat{x^0y^2z^0})_{iN_c} & \omega_{N_c}(\widehat{x^0y^0z^2})_{iN_c} & \omega_{N_c}(\widehat{x^0y^1z^1})_{iN_c} \end{pmatrix}_{N_c \times 5}, \\
\tilde{\mathbf{L}}_2 &= \begin{pmatrix} 0 & -\omega_1(\widehat{x^1y^0z^0})_{i1} & 0 & 0 & -2\omega_1(\widehat{x^1y^1z^0})_{i1} & 0 & -\frac{1}{2}\omega_1(\widehat{x^2y^0z^0})_{i1} & -\omega_1(\widehat{x^1y^0z^1})_{i1} & 0 \\ 0 & -\omega_2(\widehat{x^1y^0z^0})_{i2} & 0 & 0 & -2\omega_2(\widehat{x^1y^1z^0})_{i2} & 0 & -\frac{1}{2}\omega_2(\widehat{x^2y^0z^0})_{i2} & -\omega_2(\widehat{x^1y^0z^1})_{i2} & 0 \\ \vdots & \vdots \\ 0 & -\omega_{N_c}(\widehat{x^1y^0z^0})_{iN_c} & 0 & 0 & -2\omega_{N_c}(\widehat{x^1y^1z^0})_{iN_c} & 0 & -\frac{1}{2}\omega_{N_c}(\widehat{x^2y^0z^0})_{iN_c} & -\omega_{N_c}(\widehat{x^1y^0z^1})_{iN_c} & 0 \end{pmatrix}_{N_c \times 9}, \\
\tilde{\mathbf{L}}_3 &= \begin{pmatrix} 0 & 0 & -\omega_1(\widehat{x^1y^1z^0})_{i1} & 0 & 0 & -2\omega_1(\widehat{x^1y^0z^1})_{i1} & 0 & -\omega_1(\widehat{x^1y^1z^0})_{i1} & -\frac{1}{2}\omega_1(\widehat{x^2y^0z^0})_{i1} \\ 0 & 0 & -\omega_2(\widehat{x^1y^0z^0})_{i2} & 0 & 0 & -2\omega_2(\widehat{x^1y^0z^1})_{i2} & 0 & -\omega_2(\widehat{x^1y^1z^0})_{i2} & -\frac{1}{2}\omega_2(\widehat{x^2y^0z^0})_{i2} \\ \vdots & \vdots \\ 0 & 0 & -\omega_{N_c}(\widehat{x^1y^0z^0})_{iN_c} & 0 & 0 & -2\omega_{N_c}(\widehat{x^1y^0z^1})_{iN_c} & 0 & -\omega_{N_c}(\widehat{x^1y^1z^0})_{iN_c} & -\frac{1}{2}\omega_{N_c}(\widehat{x^2y^0z^0})_{iN_c} \end{pmatrix}_{N_c \times 9},
\end{aligned}$$

$$\tilde{\mathbf{D}}^{B_x} = (D_{010}^{B_x} \ D_{001}^{B_x} \ D_{020}^{B_x} \ D_{002}^{B_x} \ D_{011}^{B_x})^T.$$

Again, ω_i and N_c have the same definitions as in Equation (9). We are now in a position to solve system (18) in the least-squares sense by the singular value decomposition and then achieve $\tilde{\mathbf{D}}^{B_x}$, \mathbf{D}^{B_y} , \mathbf{D}^{B_z} . Afterward, we get $\frac{\partial B_x}{\partial x}$, $\frac{\partial^2 B_x}{\partial x^2}$, $\frac{\partial^2 B_x}{\partial x \partial y}$, $\frac{\partial^2 B_x}{\partial x \partial z}$ by back substitution via Equation (16). Analogous to Equation (10), the reconstructed polynomial for \mathcal{X} is constrained as

$$\begin{aligned}
\mathcal{X}_i(\mathbf{x}) &= \bar{\mathcal{X}}_i + \phi_i^{\mathcal{X}} \left(\sum_{\substack{p_1=0 \\ 0 < p_1 + p_2 + p_3 \leq 2}}^2 \sum_{\substack{p_2=0 \\ 0 < p_1 + p_2 + p_3 \leq 2}}^2 \sum_{\substack{p_3=0 \\ 0 < p_1 + p_2 + p_3 \leq 2}}^2 [(x - x_i)^{p_1} (y - y_i)^{p_2} \right. \\ &\quad \left. \times (z - z_i)^{p_3} - \overline{(x^{p_1} y^{p_2} z^{p_3})}_i \right] D_{p_1 p_2 p_3}^{\mathcal{X}} \Big), \quad (19)
\end{aligned}$$

with $\phi_i^{\mathcal{X}}$ the corresponding Venkatakrishnan limiter function defined in Section 3.2. It is worth mentioning that for K -exact polynomial reconstruction, more derivative constraints $\frac{\partial^{p_1+p_2+p_3}}{\partial x^{p_1} \partial y^{p_2} \partial z^{p_3}} (\nabla \cdot \mathbf{B}) = 0$ with $0 \leq p_1 + p_2 + p_3 \leq K - 1$ can be used in the same way as Equation (16). Up to now we have finished our first reconstructed procedure given by Equation (19), hereafter called the LSP method.

In the LSP method, the exactness of the locally divergence-free condition is kept, since the magnetic divergence, $\nabla \cdot \mathbf{B}$, at the centroid of computational cell i is directly constrained. However, in the finite-volume method, the magnetic

divergence is commonly evaluated by

$$\overline{(\nabla \cdot \mathbf{B})}_i = \frac{1}{V_i} \sum_j \mathbf{B}_{ij} \cdot \mathbf{n}_{ij} A_{ij}, \quad (20)$$

where \mathbf{B}_{ij} is the arithmetic average of $\mathbf{B}_{ij,L}$ extrapolated from the cell i and $\mathbf{B}_{ij,R}$ from the neighboring cell j , and $\mathbf{n}_{ij} = (n_{ij,x}, n_{ij,y}, n_{ij,z})^T$ is the unit face normal vector pointing from cell i to cell j . To set $\overline{(\nabla \cdot \mathbf{B})}_i = 0$ in Equation (20) provides us another option to globally constrain the magnetic divergence. This consideration motivates us to introduce our second reconstructed procedure by taking account of $\overline{(\nabla \cdot \mathbf{B})}_i = 0$, which hereafter is called the GSP method. To describe this procedure, for simplicity, here we use the following Taylor expansion to reconstruct the magnetic field:

$$\begin{aligned}
\mathcal{X}_i(\mathbf{x}) &= \bar{\mathcal{X}}_i + D_{100}^{\mathcal{X}}(x - x_i) + D_{010}^{\mathcal{X}}(y - y_i) + D_{001}^{\mathcal{X}}(z - z_i), \\
\mathcal{X} &\in \{B_x, B_y, B_z\}, \quad (21)
\end{aligned}$$

where $D_{p_1 p_2 p_3}^{\mathcal{X}} = \frac{\partial^{p_1+p_2+p_3} \mathcal{X}}{\partial x^{p_1} \partial y^{p_2} \partial z^{p_3}} \Big|_{\mathbf{x}_i}$ are the derivatives. Obviously, from Equations (20) and (21) it follows that

$$\begin{aligned}
\overline{(\nabla \cdot \mathbf{B})}_i &= \frac{1}{2V_i} \sum_j [\bar{\mathbf{B}}_i + (\nabla \otimes \mathbf{B})_i \cdot (\mathbf{x}_{ij} - \mathbf{x}_i) \\ &\quad + \bar{\mathbf{B}}_j + (\nabla \otimes \mathbf{B})_j \cdot (\mathbf{x}_{ij} - \mathbf{x}_j)] \cdot \mathbf{n}_{ij} A_{ij},
\end{aligned}$$

with j denoting the face neighbor and

$$(\nabla \otimes \mathbf{B})_i = \begin{pmatrix} D_{100}^{B_x} & D_{100}^{B_y} & D_{100}^{B_z} \\ D_{010}^{B_x} & D_{010}^{B_y} & D_{010}^{B_z} \\ D_{001}^{B_x} & D_{001}^{B_y} & D_{001}^{B_z} \end{pmatrix}.$$

Now, maintaining $(\nabla \cdot \mathbf{B})_i = 0$ means that

$$\sum_j [\bar{\mathbf{B}}_i + (\nabla \otimes \mathbf{B})_i \cdot (\mathbf{x}_{ij} - \mathbf{x}_i) + \bar{\mathbf{B}}_j + (\nabla \otimes \mathbf{B})_j \cdot (\mathbf{x}_{ij} - \mathbf{x}_j)] \cdot \mathbf{n}_{ij} A_{ij} = 0 \quad (22)$$

holds. Then, by adding Equation (22) to the usual least-squares problem about Equation (21) related to B_x , B_y , and B_z , we get the following overdetermined system:

$$\begin{pmatrix} \mathbf{L} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{L} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{L} \\ \mathbf{C}_{11} & \mathbf{C}_{12} & \mathbf{C}_{13} \end{pmatrix}_{(3N_c+1) \times 9} \begin{pmatrix} \mathbf{D}^{B_x} \\ \mathbf{D}^{B_y} \\ \mathbf{D}^{B_z} \end{pmatrix}_{9 \times 1} = \begin{pmatrix} \mathbf{R}^{B_x} \\ \mathbf{R}^{B_y} \\ \mathbf{R}^{B_z} \\ \mathbf{R}^B \end{pmatrix}_{(3N_c+1) \times 1}, \quad (23)$$

with

$$\mathbf{L} = \begin{pmatrix} \omega_1(x_1 - x_i) & \omega_1(y_1 - y_i) & \omega_1(z_1 - z_i) \\ \omega_2(x_2 - x_i) & \omega_2(y_2 - y_i) & \omega_2(z_2 - z_i) \\ \vdots & \vdots & \vdots \\ \omega_{N_c}(x_{N_c} - x_i) & \omega_{N_c}(y_{N_c} - y_i) & \omega_{N_c}(z_{N_c} - z_i) \end{pmatrix}_{N_c \times 3},$$

$$\mathbf{C}_{11} = \left(\sum_j (x_{ij} - x_i) n_{ij,x} A_{ij} \sum_j (y_{ij} - y_i) n_{ij,y} A_{ij} \times \sum_j (z_{ij} - z_i) n_{ij,z} A_{ij} \right),$$

$$\mathbf{C}_{12} = \left(\sum_j (x_{ij} - x_i) n_{ij,y} A_{ij} \sum_j (y_{ij} - y_i) n_{ij,y} A_{ij} \times \sum_j (z_{ij} - z_i) n_{ij,z} A_{ij} \right),$$

$$\mathbf{C}_{13} = \left(\sum_j (x_{ij} - x_i) n_{ij,z} A_{ij} \sum_j (y_{ij} - y_i) n_{ij,z} A_{ij} \times \sum_j (z_{ij} - z_i) n_{ij,z} A_{ij} \right),$$

$$\tilde{\mathbf{L}}_1 = \begin{pmatrix} \omega_1 \left((y_1 - y_i) - \frac{\sum_j (y_{ij} - y_i) n_{ij,x} A_{ij}}{\sum_j (x_{ij} - x_i) n_{ij,x} A_{ij}} (x_1 - x_i) \right) & \omega_1 \left((z_1 - z_i) - \frac{\sum_j (z_{ij} - z_i) n_{ij,x} A_{ij}}{\sum_j (x_{ij} - x_i) n_{ij,x} A_{ij}} (x_1 - x_i) \right) \\ \omega_2 \left((y_2 - y_i) - \frac{\sum_j (y_{ij} - y_i) n_{ij,x} A_{ij}}{\sum_j (x_{ij} - x_i) n_{ij,x} A_{ij}} (x_2 - x_i) \right) & \omega_2 \left((z_2 - z_i) - \frac{\sum_j (z_{ij} - z_i) n_{ij,x} A_{ij}}{\sum_j (x_{ij} - x_i) n_{ij,x} A_{ij}} (x_2 - x_i) \right) \\ \vdots & \vdots \\ \omega_{N_c} \left((y_{N_c} - y_i) - \frac{\sum_j (y_{ij} - y_i) n_{ij,x} A_{ij}}{\sum_j (x_{ij} - x_i) n_{ij,x} A_{ij}} (x_{N_c} - x_i) \right) & \omega_{N_c} \left((z_{N_c} - z_i) - \frac{\sum_j (z_{ij} - z_i) n_{ij,x} A_{ij}}{\sum_j (x_{ij} - x_i) n_{ij,x} A_{ij}} (x_{N_c} - x_i) \right) \end{pmatrix}_{N_c \times 2},$$

$$\mathbf{D}^{B_x} = \begin{pmatrix} D_{100}^{B_x} & D_{010}^{B_x} & D_{001}^{B_x} \end{pmatrix}^T,$$

$$\mathbf{D}^{B_y} = \begin{pmatrix} D_{100}^{B_y} & D_{010}^{B_y} & D_{001}^{B_y} \end{pmatrix}^T,$$

$$\mathbf{D}^{B_z} = \begin{pmatrix} D_{100}^{B_z} & D_{010}^{B_z} & D_{001}^{B_z} \end{pmatrix}^T,$$

$$\mathbf{R}^{B_x} = \begin{pmatrix} \omega_1(\bar{B}_{x1} - \bar{B}_{xi}) \\ \omega_2(\bar{B}_{x2} - \bar{B}_{xi}) \\ \vdots \\ \omega_{N_c}(\bar{B}_{xN_c} - \bar{B}_{xi}) \end{pmatrix}, \quad \mathbf{R}^{B_y} = \begin{pmatrix} \omega_1(\bar{B}_{y1} - \bar{B}_{yi}) \\ \omega_2(\bar{B}_{y2} - \bar{B}_{yi}) \\ \vdots \\ \omega_{N_c}(\bar{B}_{yN_c} - \bar{B}_{yi}) \end{pmatrix},$$

$$\mathbf{R}^{B_z} = \begin{pmatrix} \omega_1(\bar{B}_{z1} - \bar{B}_{zi}) \\ \omega_2(\bar{B}_{z2} - \bar{B}_{zi}) \\ \vdots \\ \omega_{N_c}(\bar{B}_{zN_c} - \bar{B}_{zi}) \end{pmatrix},$$

$$\mathbf{R}^B = - \sum_j [\bar{\mathbf{B}}_i + \bar{\mathbf{B}}_j + (\nabla \otimes \mathbf{B})_j \cdot (\mathbf{x}_{ij} - \mathbf{x}_j)] \cdot \mathbf{n}_{ij} A_{ij}.$$

Obviously, the last row in Equation (23) comes from Equation (22). Similar to the overdetermined system (17) in the LSP method, the overdetermined system (23) is solved by the elimination method so that the last equation in Equation (23) can be satisfied as accurately as possible. It can be seen that the last equation is of implicitness in that the derivatives at the centroid of surrounding cell j , $(\nabla \otimes \mathbf{B})_j$, in \mathbf{R}^B are needed when obtaining $(\nabla \otimes \mathbf{B})_i$, and thus an iterative procedure is a must. Here we first utilize $D_{100}^{B_x}$ for such elimination to obtain a reduced overdetermined system

$$\begin{pmatrix} \tilde{\mathbf{L}}_1 & \tilde{\mathbf{L}}_2 & \tilde{\mathbf{L}}_3 \\ \mathbf{0} & \mathbf{L} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{L} \end{pmatrix}_{3N_c \times 8} \begin{pmatrix} \tilde{\mathbf{D}}^{B_x} \\ \mathbf{D}^{B_y} \\ \mathbf{D}^{B_z} \end{pmatrix}_{8 \times 1} = \begin{pmatrix} \tilde{\mathbf{R}}^{B_x} \\ \mathbf{R}^{B_y} \\ \mathbf{R}^{B_z} \end{pmatrix}_{3N_c \times 1}, \quad (24)$$

with

$$\tilde{\mathbf{L}}_2 = \begin{pmatrix} -\omega_1 \frac{\sum_j (x_{ij} - x_i) n_{ij,y} A_{ij}}{\sum_j (x_{ij} - x_i) n_{ij,x} A_{ij}} (x_1 - x_i) & -\omega_1 \frac{\sum_j (y_{ij} - y_i) n_{ij,y} A_{ij}}{\sum_j (x_{ij} - x_i) n_{ij,x} A_{ij}} (x_1 - x_i) & -\omega_1 \frac{\sum_j (z_{ij} - z_i) n_{ij,y} A_{ij}}{\sum_j (x_{ij} - x_i) n_{ij,x} A_{ij}} (x_1 - x_i) \\ -\omega_2 \frac{\sum_j (x_{ij} - x_i) n_{ij,y} A_{ij}}{\sum_j (x_{ij} - x_i) n_{ij,x} A_{ij}} (x_2 - x_i) & -\omega_2 \frac{\sum_j (y_{ij} - y_i) n_{ij,y} A_{ij}}{\sum_j (x_{ij} - x_i) n_{ij,x} A_{ij}} (x_2 - x_i) & -\omega_2 \frac{\sum_j (z_{ij} - z_i) n_{ij,y} A_{ij}}{\sum_j (x_{ij} - x_i) n_{ij,x} A_{ij}} (x_2 - x_i) \\ \vdots & \vdots & \vdots \\ -\omega_{N_c} \frac{\sum_j (x_{ij} - x_i) n_{ij,y} A_{ij}}{\sum_j (x_{ij} - x_i) n_{ij,x} A_{ij}} (x_{N_c} - x_i) & -\omega_{N_c} \frac{\sum_j (y_{ij} - y_i) n_{ij,y} A_{ij}}{\sum_j (x_{ij} - x_i) n_{ij,x} A_{ij}} (x_{N_c} - x_i) & -\omega_{N_c} \frac{\sum_j (z_{ij} - z_i) n_{ij,y} A_{ij}}{\sum_j (x_{ij} - x_i) n_{ij,x} A_{ij}} (x_{N_c} - x_i) \end{pmatrix}_{N_c \times 3},$$

$$\tilde{\mathbf{L}}_3 = \begin{pmatrix} -\omega_1 \frac{\sum_j (x_{ij} - x_i) n_{ij,z} A_{ij}}{\sum_j (x_{ij} - x_i) n_{ij,x} A_{ij}} (x_1 - x_i) & -\omega_1 \frac{\sum_j (y_{ij} - y_i) n_{ij,z} A_{ij}}{\sum_j (x_{ij} - x_i) n_{ij,x} A_{ij}} (x_1 - x_i) & -\omega_1 \frac{\sum_j (z_{ij} - z_i) n_{ij,z} A_{ij}}{\sum_j (x_{ij} - x_i) n_{ij,x} A_{ij}} (x_1 - x_i) \\ -\omega_2 \frac{\sum_j (x_{ij} - x_i) n_{ij,z} A_{ij}}{\sum_j (x_{ij} - x_i) n_{ij,x} A_{ij}} (x_2 - x_i) & -\omega_2 \frac{\sum_j (y_{ij} - y_i) n_{ij,z} A_{ij}}{\sum_j (x_{ij} - x_i) n_{ij,x} A_{ij}} (x_2 - x_i) & -\omega_2 \frac{\sum_j (z_{ij} - z_i) n_{ij,z} A_{ij}}{\sum_j (x_{ij} - x_i) n_{ij,x} A_{ij}} (x_2 - x_i) \\ \vdots & \vdots & \vdots \\ -\omega_{N_c} \frac{\sum_j (x_{ij} - x_i) n_{ij,z} A_{ij}}{\sum_j (x_{ij} - x_i) n_{ij,x} A_{ij}} (x_{N_c} - x_i) & -\omega_{N_c} \frac{\sum_j (y_{ij} - y_i) n_{ij,z} A_{ij}}{\sum_j (x_{ij} - x_i) n_{ij,x} A_{ij}} (x_{N_c} - x_i) & -\omega_{N_c} \frac{\sum_j (z_{ij} - z_i) n_{ij,z} A_{ij}}{\sum_j (x_{ij} - x_i) n_{ij,x} A_{ij}} (x_{N_c} - x_i) \end{pmatrix}_{N_c \times 3},$$

$$\tilde{\mathbf{D}}^{B_x} = \begin{pmatrix} D_{010}^{B_x} & D_{001}^{B_x} \end{pmatrix}^T,$$

$$\tilde{\mathbf{R}}^{B_x} = \begin{pmatrix} \omega_1 \left(\bar{B}_{x1} - \bar{B}_{xi} - \frac{R^B}{\sum_j (x_{ij} - x_i) n_{ij,x} A_{ij}} (x_1 - x_i) \right) \\ \omega_2 \left(\bar{B}_{x2} - \bar{B}_{xi} - \frac{R^B}{\sum_j (x_{ij} - x_i) n_{ij,x} A_{ij}} (x_2 - x_i) \right) \\ \vdots \\ \omega_{N_c} \left(\bar{B}_{xN_c} - \bar{B}_{xi} - \frac{R^B}{\sum_j (x_{ij} - x_i) n_{ij,x} A_{ij}} (x_{N_c} - x_i) \right) \end{pmatrix}.$$

Then, we iteratively solve Equation (24), i.e.,

$$\begin{pmatrix} \tilde{\mathbf{L}}_1 & \tilde{\mathbf{L}}_2 & \tilde{\mathbf{L}}_3 \\ \mathbf{0} & \mathbf{L} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{L} \end{pmatrix}_{3N_c \times 8} \begin{pmatrix} \tilde{\mathbf{D}}^{B_x} \\ \mathbf{D}^{B_y} \\ \mathbf{D}^{B_z} \end{pmatrix}_{8 \times 1}^{(k+1)} = \begin{pmatrix} \tilde{\mathbf{R}}^{B_x} \\ \mathbf{R}^{B_y} \\ \mathbf{R}^{B_z} \end{pmatrix}_{3N_c \times 1}^{(k)}, \quad (25)$$

where

$$\tilde{\mathbf{R}}^{B_x} = \begin{pmatrix} \omega_1 \left(\bar{B}_{x1} - \bar{B}_{xi} - \frac{(R^B)^{(k)}}{\sum_j (x_{ij} - x_i) n_{ij,x} A_{ij}} (x_1 - x_i) \right) \\ \omega_2 \left(\bar{B}_{x2} - \bar{B}_{xi} - \frac{(R^B)^{(k)}}{\sum_j (x_{ij} - x_i) n_{ij,x} A_{ij}} (x_2 - x_i) \right) \\ \vdots \\ \omega_{N_c} \left(\bar{B}_{xN_c} - \bar{B}_{xi} - \frac{(R^B)^{(k)}}{\sum_j (x_{ij} - x_i) n_{ij,x} A_{ij}} (x_{N_c} - x_i) \right) \end{pmatrix},$$

$$(R^B)^{(k)} = -\sum_j [\bar{\mathbf{B}}_i + \bar{\mathbf{B}}_j + (\nabla \otimes \mathbf{B})_j^{(k)} \cdot (\mathbf{x}_{ij} - \mathbf{x}_j)] \cdot \mathbf{n}_{ij} A_{ij}.$$

The specific iteration procedure runs as follows:

1. First, obtain $(\nabla \otimes \mathbf{B})_i$ for all the cells by solving the usual least-squares problem about Equation (21) related to B_x , B_y , and B_z , combined with the first formula $\nabla \cdot \mathbf{B} = 0$ in Equation (16). Then use such obtained $(\nabla \otimes \mathbf{B})_i$ as $(\nabla \otimes \mathbf{B})_i^{(0)}$ to start the iteration of Equation (25).
2. Second, solve Equation (25) in the least-squares sense by the singular value decomposition and achieve $(\tilde{\mathbf{D}}^{B_x})^{(k+1)}$, $(\mathbf{D}^{B_y})^{(k+1)}$, $(\mathbf{D}^{B_z})^{(k+1)}$. Then, we get $(D_{100}^{B_x})^{(k+1)}$ by back substitution via Equation (22), since $D_{100}^{B_x}$ is used for elimination. This operation is carried out on all the computational cells. After $(\nabla \otimes \mathbf{B})_i^{(k+1)}$ are known, we proceed to the next iteration loop.
3. Third, the iterative procedure is repeated until an appropriate condition is satisfied. Usually, there are two ways: one is to set up a control condition such as $\max_i \left(\frac{|\Delta h_i| |\nabla \cdot \mathbf{B}_i|}{|\bar{B}_i|} \right) < \epsilon_5$, where ϵ_5 is a small parameter ($\epsilon_5 \approx 10^{-10}$ to 10^{-8}), and another is to simply prescribe the number of iterations in terms of practical situations.

In our actual computation, we find that the more iteration loops there are, the smaller the magnetic divergence errors become. Moreover, at every 50 iteration loops, there is a significant decrease in the divergence errors of the magnetic field. Therefore, under the trade-off between the computational efficiency and the degree to which the magnetic divergence errors are decreased, we adopt 100 iteration loops in our test cases. Again for efficiency, the reconstruction stencil used in the GSP method only includes the direct face neighbor cells, that is, N_c is taken to be 6 in Equations (23)–(25). In this paper, although we implement the GSP method only for the reconstruction expressions with the first-order derivatives of the magnetic field, it has the potential to be applied to the high-order reconstruction.

It should be mentioned that based on Equation (22), an approximate, globally divergence-free CG method is proposed

by Hopkins (2016). The CG method utilizes a correction term to iteratively obtain the preferred $(\nabla \otimes \mathbf{B})_i$ by minimizing some “penalty function.” However, different from the CG method, our proposed GSP method incorporates Equation (22) directly into the usual least-squares problem about Equation (21) related to B_x , B_y , and B_z and solves the resulting Equation (25) by an iterative procedure. Besides, Hopkins also described a locally divergence-free method by seeking the matrix $(\nabla \otimes \mathbf{B})_i$ constrained to the condition $(\nabla \cdot \mathbf{B})_i = 0$ in the least-squares formulation. However, our LSP method includes four constraints in Equation (16) simultaneously.

ORCID iDs

Xueshang Feng  <https://orcid.org/0000-0001-8605-2159>

References

- Abbo, L., Lionello, R., Riley, P., & Wang, Y.-M. 2015, *SoPh*, **290**, 2043
 Balsara, D. S. 2010, *JCoPh*, **229**, 1970
 Balsara, D. S. 2012, *JCoPh*, **231**, 7476
 Balsara, D. S., Dumbser, M., & Abgrall, R. 2014, *JCoPh*, **261**, 172
 Barth, T. J. 1993, *AIAA Meeting Papers*, 93-0668
 Brackbill, J., & Barnes, D. 1980, *JCoPh*, **35**, 426
 Cassak, P. A., & Shay, M. A. 2012, *SSRv*, **172**, 283
 Cranmer, S. R. 2009, *LRSF*, **6**, 3
 Dedner, A., Kemm, F., Kröner, D., et al. 2002, *JCoPh*, **175**, 645
 Evans, C. R., & Hawley, J. F. 1988, *ApJ*, **332**, 659
 Feng, X., Li, C., Xiang, C., et al. 2017, *ApJS*, **233**, 10
 Feng, X., Yang, L., Xiang, C., et al. 2010, *ApJ*, **723**, 300
 Feng, X., Zhang, M., & Zhou, Y. 2014, *ApJS*, **214**, 6
 Feng, X., Zhou, Y., & Wu, S. T. 2007, *ApJ*, **655**, 1110
 Feng, X. S., Ma, X. P., & Xiang, C. Q. 2015, *JGRA*, **120**, 10,159
 Frazin, R. A., Vásquez, A. M., Kamalabadi, F., & Park, H. 2007, *ApJL*, **671**, L201
 Fuchs, F., Mishra, S., & Risebro, N. 2009, *JCoPh*, **228**, 641
 Gombosi, T. I., de Zeeuw, D. L., Powell, K. G., et al. 2003, in *Space Plasma Simulation*, ed. J. Büchner, C. Dum, & M. Scholer (Berlin: Springer), 247
 Groth, C. P. T., De Zeeuw, D. L., Gombosi, T. I., & Powell, K. G. 2000, *JGRA*, **105**, 25053
 Harten, A. 1983, *JCoPh*, **49**, 357
 Hayes, A. P., Vourlidas, A., & Howard, R. A. 2001, *ApJ*, **548**, 1081
 Hopkins, P. F. 2016, *MNRAS*, **462**, 576
 Huang, K., Wu, H., Yu, H., & Yan, D. 2011, *IJNMF*, **65**, 1026
 Ivan, L., De Sterck, H., Northrup, S. A., & Groth, C. P. T. 2013, *JCoPh*, **255**, 205
 Ivan, L., De Sterck, H., Susanto, A., & Groth, C. P. T. 2015, *JCoPh*, **282**, 157
 Jiang, C., Feng, X., Zhang, J., & Zhong, D. 2010, *SoPh*, **267**, 463
 Jiang, G.-S., & Wu, C.-c. 1999, *JCoPh*, **150**, 561
 Levy, D. W., Powell, K. G., & van Leer, B. 1993, *JCoPh*, **106**, 201
 Li, H. C., & Feng, X. S. 2018, *JGRA*, **123**, 4488
 Linker, J. A., Mikić, Z., Biesecker, D. A., et al. 1999, *JGRA*, **104**, 9809
 Lowder, C., Qiu, J., & Leamon, R. 2017, *SoPh*, **292**, 18
 Luhmann, J. G., Li, Y., Arge, C. N., Gazis, P. R., & Ulrich, R. 2002, *JGRA*, **107**, 1154
 Luhmann, J. G., Petrie, G., & Riley, P. 2013, *JAdR*, **4**, 221
 Michalak, C., & Ollivier-Gooch, C. 2009, *JCoPh*, **228**, 8693
 Nakamizo, A., Tanaka, T., Kubo, Y., et al. 2009, *JGRA*, **114**, A07109
 Nishikawa, H., & Kitamura, K. 2008, *JCoPh*, **227**, 2560
 Orszag, S. A., & Tang, C.-M. 1979, *JFM*, **90**, 129
 Pahud, D. M., Merkin, V. G., Arge, C. N., Hughes, W. J., & McGregor, S. M. 2012, *JASTP*, **83**, 32
 Parker, E. N. 1963, *Interplanetary Dynamical Processes* (New York: Interscience)
 Pätzold, M., Tsurutani, B. T., & Bird, M. K. 1997, *JGR*, **102**, 24151
 Petrie, G. J. D., Canou, A., & Amari, T. 2011, *SoPh*, **274**, 163
 Porfir'eva, G. A., Yakunina, G. V., Delone, A. B., & Oreshina, I. V. 2009, *JPhSt*, **13**, 2901
 Powell, K. G., Roe, P. L., Linde, T. J., Gombosi, T. I., & De Zeeuw, D. L. 1999, *JCoPh*, **154**, 284
 Powell, K. G., Roe, P. L., & Quirk, J. 1993, in *Algorithmic Trends in Computational Fluid Dynamics*, ed. M. Y. Hussaini, A. Kumar, & M. D. Salas (New York: Springer), 303
 Press, W. H., Teukolsky, S. A., Vetterling, W. T., & Flannery, B. P. 1992, *Numerical Recipes in Fortran 77* (2nd ed.; Cambridge: Cambridge Univ. Press)
 Ren, Y.-X. 2003, *CF*, **32**, 1379
 Riley, P., Linker, J. A., & Arge, C. N. 2015, *SpWea*, **13**, 154
 Riley, P., Linker, J. A., Lionello, R., & Mikić, Z. 2012, *JASTP*, **83**, 1
 Riley, P., Lionello, R., Linker, J. A., et al. 2011, *SoPh*, **274**, 361
 Roe, P. 2017, in *Handbook of Numerical Analysis*, Vol. 18, *Handbook of Numerical Methods for Hyperbolic Problems*, ed. R. Abgrall & C.-W. Shu (Amsterdam: Elsevier), 53
 Roe, P. L. 1981, *JCoPh*, **43**, 357
 Roe, P. L., & Balsara, D. S. 1996, *SJAM*, **56**, 57
 Rusanov, V. V. 1961, *ZVMMF*, **1**, 267
 Serna, S. 2009, *JCoPh*, **228**, 4232
 Shen, Y., Zha, G., & Huerta, M. 2011, *AIAA Meeting Papers*, 2011-383
 Sitaraman, H., & Raja, L. L. 2013, *JCoPh*, **251**, 364
 Tanaka, T. 1994, *JCoPh*, **111**, 381
 Tang, H.-Z., & Xu, K. 2000, *JCoPh*, **165**, 69
 Tóth, G. 2000, *JCoPh*, **161**, 605
 Tran, T. 2009, PhD thesis, Univ. California, Los Angeles
 van der Holst, B., & Keppens, R. 2007, *JCoPh*, **226**, 925
 Venkatakrishnan, V. 1993, *On the Accuracy of Limiters and Convergence to Steady State Solutions*, Tech. Rep. 93-0880
 Vides, J., Nkonga, B., & Audit, E. 2015, *JCoPh*, **280**, 643
 Yalim, M., Abeele, D. V., Lani, A., Quintino, T., & Deconinck, H. 2011, *JCoPh*, **230**, 6136
 Yang, L. P., Feng, X. S., Xiang, C. Q., et al. 2012, *JGRA*, **117**, 8110
 Yang, Y., Feng, X.-S., & Jiang, C.-W. 2017, *JCoPh*, **349**, 561
 Zhang, F., Liu, J., Chen, B., & Zhong, W. 2016, *CF*, **134**, 11
 Zhang, M., John, Yu, Henry, S.-T., Lin, S.-C., Chang, S.-C., & Blankson, I. 2006, *JCoPh*, **214**, 599
 Zhang, S.-H., Feng, X. S., Wang, Y., & Yang, L.-P. 2011, *ChPhL*, **28**, 089601
 Zhou, Y., & Feng, X. 2014, *ScChE*, **57**, 153