

# On Resistive MHD Models with Adaptive Moving Meshes

PAUL A. ZEGELING

*Mathematical Institute, Utrecht University  
Budapestlaan 6, 3584 CD, Utrecht, The Netherlands  
email: zegeling@math.uu.nl*

## Abstract

In this paper we describe an adaptive moving mesh technique and its application to convection-diffusion models from magnetohydrodynamics (MHD). The method is based on a coordinate transformation between physical and computational coordinates. The transformation can be viewed as a solution of adaptive mesh partial differential equations (PDEs) which are derived from the minimization of a mesh-energy integral. For an efficient implementation we have used an approach in which the numerical solution of the physical PDE model and the adaptive PDEs are decoupled. Further, to avoid solving large nonlinear systems, an implicit-explicit method is applied for the time integration in combination with the iterative method Bi-CGSTAB. The adaptive mesh can be viewed as a 2D variant of the equidistribution principle, and it has the ability to track individual features of the physical solutions in the developing plasma flows. The results of a series of numerical experiments are presented which cover several aspects typifying resistive magnetofluid-dynamics.

**Keywords:** magnetohydrodynamics; coordinate transformation; equidistribution principle; monitor functions; adaptive moving mesh method; implicit-explicit time integration.

# 1 Introduction

Many interesting phenomena in plasma fluid dynamics can be described within the framework of magnetohydrodynamics (MHD). Numerical studies in plasma flows frequently involve simulations with highly varying spatial and temporal scales. As a consequence, numerical methods on uniform meshes are inefficient to use, since a very large number of mesh points is needed to resolve the spatial structures, such as shocks, contact discontinuities, shear layers, or current sheets. For the efficient study of these phenomena, we require adaptive mesh methods which automatically track and spatially resolve one or more of these structures.

Over the years a large number of adaptive mesh methods have been proposed for time-dependent PDE models. Two main strategies of adaptive mesh methods can be distinguished, namely, static-remeshing methods and moving-mesh or dynamic-remeshing methods. In static-remeshing methods (denoted by  $h$ -refinement) the location of nodes is fixed. A method of this type adapts the mesh by adding nodes where they are necessary and removing them when they are no longer needed. The refinement or de-refinement is controlled by error estimates or error monitor values (which have no resemblance with the true numerical error). Recent examples of these methods are described in [17], [24], [12]. In dynamic-remeshing methods (denoted by  $r$ -refinement) nodes are moving continuously in the space-time domain, like in classical Lagrangian methods, and the discretization of the PDE is coupled with the motion of the mesh. Examples can be found in [9], [16], [14], [23].

In this paper we follow the second approach. We describe an adaptive moving mesh technique that is based on a minimization of a so-called mesh-energy integral. The corresponding Euler-Lagrange equations then define a set of adaptive mesh PDEs. In one space dimension this reduces to the widely-used equidistribution principle; in two space dimensions it is related to (but not equivalent with) harmonic mapping theory supplemented with a monitor matrix to detect the steep transitions in the solution. For the integration in time of the physical PDE the first-order implicit-explicit method 1-SBDF [1] is applied. The implicit part of this method deals with the (linear) diffusion, whereas the explicit part takes care of the convection terms in the model. Additionally, the discretization of the PDE model and the moving mesh is decoupled. For solving the linear (non-symmetric) system behind the discretized equations, the iterative method Bi-CGSTAB (see [18]) is used. Since the physical domain in our application is rectangular, we have

simplified, similar to [16], the adaptive mesh PDEs as used in [3] and in [9]. Further, we have borrowed ideas from [3] in which the difficult choice of a user-defined adaptivity constant in the monitor function is replaced by a procedure where the parameter is time-dependent and automatically chosen. This simplified and robust formulation is then applied to convection-diffusion models from MHD.

The layout of the paper is as follows. In the next section we present the full set of MHD equations, their physical meaning and the restriction to a two-dimensional situation. The adaptive moving mesh is defined as the solution of a system of nonlinear diffusion PDEs in section 3. Section 4 describes the numerical algorithm for solving the physical and mesh PDEs. In section 5 numerical experiments are shown for several characteristic models from resistive MHD, viz., a model with a four-cell convection pattern, a single eddy model and a model with a band of eddies in a horizontal magnetic field. Finally, section 6 lists the conclusions and presents an outlook how to extend and further improve the method.

## 2 The equations of magnetohydrodynamics

The MHD equations govern the dynamics of a charge-neutral ‘plasma’. Just like the conservative Euler equations provide a continuum description for a compressible gas, the MHD equations express the basic physical conservation laws to which a plasma must obey. Because plasma dynamics is influenced by magnetic fields through the Lorentz-force, the needed additions in going from hydrodynamic to magnetohydrodynamic behaviour is a vector equation for the magnetic field evolution and extra terms in the Euler system that quantify the magnetic force and energy density.

Using the conservative variables density  $\rho$ , momentum density  $\mathbf{m} \equiv \rho\mathbf{v}$  (with velocity  $\mathbf{v}$ ), magnetic field  $\mathbf{B}$ , and total energy density  $e$ , the ideal MHD equations can be written as follows (cfr. [7], [22]):

Conservation of mass:

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho\mathbf{v}) = 0. \quad (1)$$

Conservation of momentum:

$$\frac{\partial(\rho\mathbf{v})}{\partial t} + \nabla \cdot (\rho\mathbf{v}\mathbf{v} - \mathbf{B}\mathbf{B}^T) + \nabla p_{tot} = 0. \quad (2)$$

Conservation of energy:

$$\frac{\partial e}{\partial t} + \nabla \cdot (e\mathbf{v} + \mathbf{v}p_{tot} - \mathbf{B}\mathbf{B}^T \cdot \mathbf{v}) = 0 \quad [+ \epsilon_m(\nabla \times \mathbf{B})^2]. \quad (3)$$

Magnetic field induction equation:

$$\frac{\partial \mathbf{B}}{\partial t} + \nabla \cdot (\mathbf{v}\mathbf{B}^T - \mathbf{B}\mathbf{v}^T) = 0 \quad [+ \epsilon_m \Delta \mathbf{B}]. \quad (4)$$

In (2) and (3) the total pressure  $p_{tot}$  consists of both a thermal and a magnetic contribution as given by

$$p_{tot} = p + \frac{\mathbf{B}^2}{2}, \quad \text{where } p = (\gamma - 1)\left(e - \rho \frac{\mathbf{v}^2}{2} - \frac{\mathbf{B}^2}{2}\right) \quad (5)$$

is the thermal pressure (notation:  $\mathbf{v}^2 = \mathbf{v}^T \mathbf{v}$  and  $\mathbf{B}^2 = \mathbf{B}^T \mathbf{B}$ ). This set of equations must be solved in conjunction with an important condition on the magnetic field  $\mathbf{B}$ , namely the non-existence of magnetic ‘charge’ or monopoles. Mathematically, it is easily demonstrated that this property can be imposed as an initial condition alone, since

$$\nabla \cdot \mathbf{B}|_{t=0} = 0 \implies \nabla \cdot \mathbf{B}|_{t \geq 0} = 0. \quad (6)$$

In multi-dimensional numerical MHD, the combined spatio-temporal discretization may not always ensure this conservation of the solenoidal character of the vector magnetic field.

The terms between brackets in equations (3) and (4) extend the ideal MHD model with the effects of Ohmic heating due to the presence of currents. With the resistivity  $\epsilon_m \neq 0$ , we then solve the resistive MHD equations.

## 2.1 2D magnetic field evolution

The core problem is represented by the induction equation (4), alternatively written as

$$\frac{\partial \mathbf{B}}{\partial t} = \nabla \times (\mathbf{v} \times \mathbf{B}) + \epsilon_m \Delta \mathbf{B} \quad (7)$$

with  $\epsilon_m > 0$  the resistivity or magnetic diffusion coefficient. As may be anticipated, the parameter  $\epsilon_m$  is related to the inverse of the *magnetic Reynolds*

*number* (also named *Lundquist number*). In two space dimensions, we set  $\mathbf{B} = (B_1, B_2, 0)$ , to obtain the following system of PDEs,

$$\frac{\partial B_1}{\partial t} = \epsilon_m \Delta B_1 + v_1 \frac{\partial B_2}{\partial y} - v_2 \frac{\partial B_1}{\partial y} + B_2 \frac{\partial v_1}{\partial y} - B_1 \frac{\partial v_2}{\partial y}, \quad (8)$$

$$\frac{\partial B_2}{\partial t} = \epsilon_m \Delta B_2 - v_1 \frac{\partial B_2}{\partial x} + v_2 \frac{\partial B_1}{\partial x} - B_2 \frac{\partial v_1}{\partial x} + B_1 \frac{\partial v_2}{\partial x}, \quad (9)$$

together with the property  $\nabla \cdot \mathbf{B} = 0$ .

One way to ensure a divergence-free magnetic field at all times is to make use of a vector potential formulation where  $\mathbf{B} \stackrel{\text{def}}{=} \nabla \times \mathbf{A}$ . From this follows automatically that  $\nabla \cdot \mathbf{B} = \nabla \cdot (\nabla \times \mathbf{A}) = 0$ . In two dimensional applications, the system (8)-(9) is then equivalent to the single PDE for the scalar  $A_3$  component

$$\frac{\partial A_3}{\partial t} = -\mathbf{v} \cdot \nabla A_3 + \epsilon_m \Delta A_3, \quad (10)$$

with  $\frac{\partial A_3}{\partial y} = B_1$ ,  $-\frac{\partial A_3}{\partial x} = B_2$ , while  $\mathbf{A} = (0, 0, A_3)^T$ . Note that magnetic field lines are isolines of this  $A_3$  potential.

Finally, we point out (cfr. [19]) that the partial problem posed by the system (8)-(9), or equivalently the PDE (10), can be relevant as a physical solution to the special case where we consider incompressible flow  $\nabla \cdot \mathbf{v} = 0$ , the momentum equation (2) under the condition that the magnetic energy  $\mathbf{B}^2/2$  is much smaller than the kinetic energy  $\rho \mathbf{v}^2/2$ , and the induction equation itself. In those circumstances, the momentum balance decouples from the magnetic field evolution. In fact, our model then merely consists of equation (4) re-written in the potential formulation (10), i.e., for instance, the momentum density does not appear in the final model. In the model problems studied, we therefore prescribe an incompressible flow field  $\mathbf{v}(x, y)$  as well.

## 2.2 Kinematic flux expulsion

This model problem dates back to 1966 [19], as one of the first studies to address the role of the magnetic field in a convecting plasma. Starting from a uniform magnetic field, its distortion by cellular convection patterns was simulated numerically for various values of the resistivity  $\epsilon_m$ .

One possible situation of 2D kinematic flux expulsion uses an imposed four-cell convection pattern with its incompressible velocity field given by

$$\mathbf{v} = (-\sin(2\pi x) \cos(2\pi y), \cos(2\pi x) \sin(2\pi y))^T. \quad (11)$$

We then want to solve for the scalar vector potential  $A_3$  from (10) on the domain  $(x, y) \in [0, 1] \times [0, 1]$  and for times  $t \in [0, 5]$ . In terms of  $A_3$ , the initial uniform vertical field is obtained through  $A_3|_{t=0} = 1 - x$ , while the boundary conditions are  $A_3|_{x=0} = 1$ ,  $A_3|_{x=1} = 0$ ,  $\frac{\partial A_3}{\partial n}|_{y=0} = \frac{\partial A_3}{\partial n}|_{y=1} = 0$ , which corresponds with a constant initial magnetic field.

### 2.3 Asymptotic analysis of the model

Since  $\epsilon_m$  is assumed to be small, we write the formal asymptotic expansion  $A_3(x, y, t) = \sum_{j=0}^{\infty} a_{3,j}(x, y, t) \epsilon_m^j$ . First, only considering the first-order term in the expansion, gives the hyperbolic equation

$$\frac{\partial a_{3,0}}{\partial t} = -\mathbf{v} \cdot \nabla a_{3,0}. \quad (12)$$

It follows that the initial solution remains constant on the sub-characteristics of (10) defined by (12):  $(\dot{x}, \dot{y})^T = \mathbf{v}$ . As  $\nabla \cdot \mathbf{v} = 0$ , the only critical points of this ODE system are centers or saddle points. The situation is sketched in Figure 6 (left panel) and in MHD terms this phase is called field amplification. At some point of time this situation is changed dramatically, because the solution  $a_{3,0}$  can not satisfy the boundary conditions of the original PDE model. Thus, the approximation (12) can not hold anymore and this means that boundary (and internal) layers are introduced (see Figure 6, right panel): the magnetic flux concentrates at edges of convective cells. It can be shown (see [15]) that the layers, i.e. where the term  $\epsilon_m \Delta A_3$  comes into play, have width  $\mathcal{O}(\sqrt{\epsilon_m})$ . For  $t \rightarrow \infty$  the solution reaches a non-trivial steady-state, in which the diffusion and convection terms settle down to an equilibrium. The described situation is typical for the cases in resistive MHD as discussed in this paper.

## 3 The adaptive moving mesh method

### 3.1 A coordinate transformation

Consider the scalar convection-diffusion model (10). It is common and useful in structured adaptive mesh methods to first apply a coordinate transformation. The adaptive mesh can then be seen as a uniform discretization of this mapping in the new variables. Applying the transformation

$$\xi = \xi(x, y, t), \quad \eta = \eta(x, y, t), \quad \theta = t, \quad (13)$$

to equation (10) gives [13] (a similar derivation can be made for the  $(B_1, B_2)$  system)

$$\begin{aligned} & \mathcal{J}A_{3,\theta} + A_{3,\xi}(x_\eta y_\theta - x_\theta y_\eta) + A_{3,\eta}(x_\theta y_\xi - x_\xi y_\theta) \\ & = A_{3,\xi}(-v_1 y_\eta + v_2 x_\eta) + A_{3,\eta}(v_1 y_\xi - v_2 x_\xi) \\ & + \epsilon_m \left[ \left( \frac{x_\eta^2 + y_\eta^2}{\mathcal{J}} A_{3,\xi} \right)_\xi - \left( \frac{x_\xi x_\eta + y_\xi y_\eta}{\mathcal{J}} A_{3,\eta} \right)_\xi - \left( \frac{x_\xi x_\eta + y_\xi y_\eta}{\mathcal{J}} A_{3,\xi} \right)_\eta + \left( \frac{x_\xi^2 + y_\xi^2}{\mathcal{J}} A_{3,\eta} \right)_\eta \right]. \end{aligned} \quad (14)$$

To show how to obtain equation (14), we work out a few terms. For example, using the chain rule of differentiation,

$$A_{3,t} = A_{3,\theta}\theta_t + A_{3,\xi}\xi_t + A_{3,\eta}\eta_t.$$

We can also find that

$$\xi_t = -x_\theta \xi_x - y_\theta \xi_y = -\frac{1}{\mathcal{J}}(x_\theta y_\eta - y_\theta x_\eta).$$

Using a similar relation for  $\eta_t$  gives us

$$A_{3,t} = A_{3,\theta} + \frac{A_{3,\xi}}{\mathcal{J}}(x_\eta y_\theta - x_\theta y_\eta) + \frac{A_{3,\eta}}{\mathcal{J}}(x_\theta y_\xi - x_\xi y_\theta).$$

Since  $\xi_x = \frac{y_\eta}{\mathcal{J}}$ ,  $\xi_y = -\frac{x_\eta}{\mathcal{J}}$ ,  $\eta_y = \frac{x_\xi}{\mathcal{J}}$  and  $\eta_x = -\frac{y_\xi}{\mathcal{J}}$ , we find for the first-order spatial derivative terms in (10):

$$A_{3,x} = \frac{1}{\mathcal{J}}(A_{3,\xi}y_\eta - A_{3,\eta}y_\xi)$$

and

$$A_{3,y} = \frac{1}{\mathcal{J}}(A_{3,\eta}x_\xi - A_{3,\xi}x_\eta).$$

With these results the Laplacian term in (10) can also be transformed in a rather straightforward manner to the form in equation (10). Note that,  $\mathcal{J} = x_\xi y_\eta - x_\eta y_\xi$  denotes the determinant of the Jacobian of the inverse transformation.

### 3.2 The adaptive mesh PDEs

The transformation, in other words, the adaptive mesh, is prescribed by so-called adaptive moving mesh PDEs.

For simplicity, let us suppress the time-dependence of the transformation: we will use the physical coordinates  $(x, y)^T \in \Omega_p$  and the computational coordinates  $(\xi, \eta)^T \in \Omega_c := [0, 1] \times [0, 1]$ , respectively. In a variational setting, a ‘mesh-energy’ functional is defined by

$$\tilde{\mathcal{E}}(\xi, \eta) = \frac{1}{2} \int \int_{\Omega_p} \left( \tilde{\nabla} \xi^T N_1^{-1} \tilde{\nabla} \xi + \tilde{\nabla} \eta^T N_2^{-1} \tilde{\nabla} \eta \right) dx dy, \quad (15)$$

where  $\tilde{\nabla} \stackrel{\text{def}}{=} \left( \frac{\partial}{\partial x}, \frac{\partial}{\partial y} \right)^T$  and  $N_1, N_2$  are given monitor matrices. The mesh, i.e. the transformation, is determined by minimizing the energy functional via the Euler-Lagrange equations:  $\tilde{\nabla} \cdot (N_1^{-1} \tilde{\nabla} \xi) = 0$ ,  $\tilde{\nabla} \cdot (N_2^{-1} \tilde{\nabla} \eta) = 0$ .

A simple choice for the monitor matrices is  $N_1 = N_2 = \omega I$ , where  $I$  is the identity matrix and  $\omega$  a positive weight function. With this choice we obtain Winslow’s variable diffusion method [20]:

$$\tilde{\nabla} \cdot \left( \frac{1}{\omega} \tilde{\nabla} \xi \right) = 0, \quad \tilde{\nabla} \cdot \left( \frac{1}{\omega} \tilde{\nabla} \eta \right) = 0. \quad (16)$$

In one space dimension on  $\Omega_p = [x_l, x_r]$ , equations (16) reduce to  $(\omega^{-1} \xi_x)_x = 0$ , which gives the *equidistribution principle*:  $\omega^{-1} \xi_x = c(\text{onstant}) \Leftrightarrow \omega x_\xi = \tilde{c}(\text{onstant})$ , or equivalently the boundary-value problem:  $(\omega x_\xi)_\xi = 0$ , with boundary conditions  $x(0) = x_l, x(1) = x_r$ . For this case, an explicit formula for the transformation  $\xi(x)$  can be derived. Note that

$$1 = \xi(x_r) - \xi(x_l) = \int_{x_l}^{x_r} \xi_x dx = c \int_{x_l}^{x_r} \omega d\bar{x} \Rightarrow \xi_x = \frac{\omega(x)}{\int_{x_l}^{x_r} \omega d\bar{x}},$$

and integrating once more gives  $\xi(x) = \frac{\int_{x_l}^x \omega(\bar{x}) d\bar{x}}{\int_{x_l}^{x_r} \omega(\bar{x}) d\bar{x}}$ .

The ‘energy’ in (15) can be taken to represent the energy of a system of springs with spring constants  $\omega_{i,j}$  spanning the 2D mesh. Mesh points  $(x_{i,j}, y_{i,j})$  can then be seen as the mass points of the spring system, that will be ‘optimally’ positioned by minimizing the energy (see also Figure 6). For more general choices of  $N_1$  and  $N_2$  in equation (15), however, it is not obvious how to link this formulation in terms of springs and masses to mesh points and monitor values.



In practice, the physical domain may have very complex geometry and as a result directly solving the elliptic system (16) on structured meshes is unrealistic. Therefore, usually the corresponding mesh generation equations on the computational domain are solved by interchanging the dependent and independent variables. However, the resulting system is much more complicated than the original equations (16), which requires additional computational effort in obtaining numerical approximations. If the physical domain is convex, an alternative approach (see also [16]) is to consider the energy functional in the computational domain. Unlike the re-written classical equations (16) a very simple structure is maintained. In 2-D the functional now reads:

$$\mathcal{E}(x, y) = \frac{1}{2} \int \int_{\Omega_c} (\nabla^T x M_1 \nabla x + \nabla^T y M_2 \nabla y) \, d\xi d\eta,$$

where  $\nabla \stackrel{\text{def}}{=} (\frac{\partial}{\partial \xi}, \frac{\partial}{\partial \eta})^T$ . The corresponding Euler-Lagrange equations with the special choice  $M_1 = M_2 = \omega I$  (à la Winslow) are then of the form

$$\begin{aligned} (\omega x_\xi)_\xi + (\omega x_\eta)_\eta &= 0, \\ (\omega y_\xi)_\xi + (\omega y_\eta)_\eta &= 0. \end{aligned} \tag{17}$$

These elliptic equations form the basis of the adaptive mesh algorithm. Note that other choices for  $M_1$  and  $M_2$ , or even extended versions of the energy functional are well possible. We choose for the simplest and most effective one that is suitable for our application. For the interested reader, we refer to [8] or [9] for more details on other types of monitors and functionals.

### 3.3 Theoretical results

In this section a few theoretical results are presented concerning the regularity of the elliptic system (17). It must first be noted that the current method is strongly related to the harmonic mapping method as mentioned, e.g., in [4]. However, there is a slight subtle difference in the energy functional from harmonic mapping theory compared to our situation, so that we cannot simply apply the theoretical results on the regularity and unicity for harmonic maps. Fortunately, two other results in this respect can be stated, as follows:

#### **The relation of the method with equidistribution:**

Under an extra assumption on the transformation, the relation between the elliptic system (17) and the equidistribution principle becomes clear. How-

ever, it is not obvious how this condition translates to situations from practice or how to interpret it physically. In some sense, it is, strangely enough, weakly linked with a kind of derivative type of constraint on the Jacobian  $\mathcal{J} = x_\xi y_\eta - x_\eta y_\xi$ .

**Lemma 3.1** *If  $\nabla(x_\xi) \cdot \nabla(y_\eta) - \nabla(x_\eta) \cdot \nabla(y_\xi) = 0$ , then the adaptive transformation which solves (17) satisfies  $\mathcal{J}\omega = \text{constant}$ .*

Proof: First, expand and re-arrange equations (17), to get

$$\frac{1}{\omega} \nabla \omega \cdot \nabla x = -\Delta x, \quad \frac{1}{\omega} \nabla \omega \cdot \nabla y = -\Delta y.$$

It is useful to work out  $\Delta \mathcal{J}$ . For this purpose, set  $\bar{d} \stackrel{\text{def}}{=} -\ln(\omega)$ . Then after some calculus it follows that  $\Delta \mathcal{J} = \mathcal{J} \Delta \bar{d} + \nabla \bar{d} \cdot \nabla \mathcal{J} + R$ , with  $R = 2[\nabla(x_\xi) \cdot \nabla(y_\eta) - \nabla(x_\eta) \cdot \nabla(y_\xi)]$ . With  $\bar{\mathcal{J}} \stackrel{\text{def}}{=} \ln(\mathcal{J})$  we get  $\nabla \mathcal{J} = \mathcal{J} \nabla \bar{\mathcal{J}}$ . From the assumption follows  $R = 0$ . Combining these results gives  $\Delta(\bar{\mathcal{J}} - \bar{d}) + \nabla \bar{\mathcal{J}} \cdot \nabla(\bar{\mathcal{J}} - \bar{d}) = 0$ . A solution to this equation is  $\nabla(\bar{\mathcal{J}} - \bar{d}) = \mathbf{0}$ . Integrating once, we obtain  $\bar{\mathcal{J}} - \bar{d} = \text{constant}$ . This is equivalent with  $\ln(\mathcal{J}) + \ln(\omega) = \text{constant}$ . Thus  $\mathcal{J}\omega = \text{constant}$ , which relates (17) to the equidistribution principle.

### A result from differential geometry:

Without a proof we give the important theoretical result by Clément et al [6] on the non-singularity of the transformation satisfying (17). Their analysis confirms the result from harmonic mapping theory mentioned earlier. The proof consists of three main ingredients: application of the Carleman-Hartman-Winter theorem, the Jordan curve theorem and the maximum principle for elliptic PDEs. Note that in three space dimensions the application of the first two theorems is not clear. The 2D result reads:

**Lemma 3.2** *Let  $\omega \geq c > 0$ ,  $\omega \in C^{0,1}(\Omega_c)$  and  $\omega_\xi, \omega_\eta \in C^\gamma(\bar{\Omega}_c)$ ,  $\gamma \in (0, 1)$ . With the boundary conditions  $x|_{\xi=0} = y|_{\eta=0} = 0, x|_{\xi=1} = y|_{\eta=1} = 1, \frac{\partial x}{\partial n}|_{\xi=0} = \frac{\partial x}{\partial n}|_{\xi=1} = \frac{\partial y}{\partial n}|_{\eta=0} = \frac{\partial y}{\partial n}|_{\eta=1} = 0$ , there exists a unique solution  $(x, y) \in C^2(\Omega_c)$ , which is a bijection from  $\bar{\Omega}_c$  into itself. Moreover, the Jacobian  $\mathcal{J} > 0$ .*

### 3.4 Choice of the monitor function $\omega$

A traditional choice for the monitor function to detect regions with high spatial activity is an arclength-type monitor

$$\omega = \sqrt{1 + \beta \nabla A_3 \cdot \nabla A_3}. \quad (18)$$

Instead of (18), many other choices for the monitor function are possible, for example using the curvature of the solution. Of course, this type of function would generally perform better in regions of high curvature, but would also be more difficult to implement for practical applications. The choice of the monitor function  $\omega$  is an important topic in adaptive meshes. Recently, in [8] an interesting study was done on more sophisticated monitors and their relation with error estimates. We have chosen for a simple alternative approach, yet related to equation (18).

The parameter  $\beta$  in (18) is an ‘adaptivity’-parameter which controls the amount of adaptivity. For  $\beta = 0$ , we get  $\omega = 1$ . Equations (17) then yield a system of two Laplace equations for the mesh with trivial boundary conditions on the unit square. The solution of this system, obviously, is the identity transformation  $\xi(x, y) = x$ ,  $\eta(x, y) = y$ , representing a uniform mesh in both directions. Higher values of  $\beta > 0$  allow for more adaptivity. However,  $\beta$  is problem-dependent: in general, there is no straightforward rule how to choose this parameter. The situation is even worse: from literature it is known that several experiments have to be done to arrive at an ‘optimal’ choice w.r.t. accuracy and efficiency.

A recently proposed alternative for this monitor function (see also [2], [3]) deals with this issue and involves a *time-dependent* parameter that is automatically chosen. In the MHD-application of this paper it reads

$$\omega = \beta(t) + \sqrt{\nabla A_3 \cdot \nabla A_3}, \quad \text{with } \beta(t) = \int_0^1 \int_0^1 \sqrt{\nabla A_3 \cdot \nabla A_3} \, d\xi d\eta. \quad (19)$$

Since  $\frac{\partial A_3}{\partial y} = B_1$  and  $\frac{\partial A_3}{\partial x} = -B_2$ , we have  $\beta(t) = \int_0^1 \int_0^1 \sqrt{B_1^2 + B_2^2} \, d\xi d\eta = \int_0^1 \int_0^1 \mathbf{B} \, d\xi d\eta$ , which is related to the *total* magnetic energy in the model.

A justification for this choice can be given following a 1D-argument (see also [2], but note that we use computational derivatives instead of physical derivatives in this paper). In that case, we get  $\omega = \beta(t) + \sqrt{(A_x)^2} = \beta(t) + |A_x|$ ,  $\beta(t) = \int_{x_l}^{x_r} |A_x(s, t)| \, ds$ , where  $A(x, t)$  denotes the PDE solution in 1D. The monitor serves to concentrate mesh points near high first-order

derivatives of the solution. However, we do not want too many mesh points in the steep layer, because this negatively influences the numerical approximations of derivatives, as can be analyzed by investigation of the truncation errors [21]. It can be derived that, outside the layer, the monitor satisfies approximately (assuming  $|u_x| \approx 0$ ):  $\xi_x \approx 1/2$ . The quantity  $\xi_x$  represents the meshpoint concentration. Thus, approximately 50% of the mesh points will be located outside layer and we have, therefore, a smoother distribution of the mesh. But the main message is that the adaptivity parameter is chosen automatically, i.e., without tuning. It must be noted that in the paper [10] a mathematically more elegant derivation of  $\beta(t)$  is given.

## 4 Numerical solution of the PDE system

One approach, which is efficient in 1D, would be to couple the discretized systems for the adaptive mesh PDEs and the physical PDE. However, there are a number of disadvantages to this approach. First, the size of the resulting system in higher space dimensions would be large and even for moderate grid densities may be prohibitive. Second, this approach does not easily admit different convergence criteria for the mesh and physical solution. Further, it is not necessary to compute the mesh with the same level of accuracy as the physical solution. Finally, a user may wish to control over the discretization of the physical problem and such flexibility is severely restricted by coupling the unknowns together into one large nonlinear system of equations. We have therefore decoupled the numerical solution procedure for the physical and adaptive mesh PDEs, and integrate in time in an iterative manner, solving for the mesh and the physical solution alternately. Furthermore, instead of solving (17) we integrate in time the parabolic PDE system

$$\begin{aligned} x_\tau &= (\omega x_\xi)_\xi + (\omega x_\eta)_\eta, \\ y_\tau &= (\omega y_\xi)_\xi + (\omega y_\eta)_\eta, \end{aligned} \tag{20}$$

where  $\tau$  is an artificial time variable within the time integration process. In the theoretical limit,  $\tau \rightarrow \infty$ , the mesh reaches the steady-state situation (17). Numerically this means that after a number of time steps  $\Delta\tau$  the mesh will adjust to the physical PDE solution. The decoupled procedure, which is related to the alternate solution procedure in [9], is outlined in Algorithm 1. Within the decoupled procedure we freeze the coefficients in system (20) and replace the spatial derivatives by second-order central difference operators.

---

**Algorithm 1** The decoupled numerical PDE procedure

---

Given the physical solution  $A_3^{(n)}$ , the mesh  $\mathbf{x}^{(n)}$  and the time stepsize  $\Delta t$  at time  $t = t_n$ .

- 1: Calculate the new monitor function  $\omega^{(n)} = \omega^{(n)}(\mathbf{x}^{(n)}, A_3^{(n)})$ .
  - 2: Calculate the new mesh  $\mathbf{x}^{(n+1)}$  by integrating (20) from  $t = t_n$  to  $t = t_n + \Delta t$ , using  $\mathbf{x}^{(n)}$  as initial mesh and keeping the monitor function  $\omega$  constant in time during the integration. In this time step  $\Delta t$  the adaptive mesh PDEs are numerically integrated with a number of internal time steps  $\Delta\tau$  until the difference in the computed values is below some tolerance value.
  - 3: Calculate the physical solution  $A_3^{(n+1)}$  by integrating the physical PDE from  $t = t_n$  to  $t = t_n + \Delta t$ , using the mesh  $\mathbf{x}^{(n+1)}$  and mesh speed  $(x_\theta, y_\theta)^T \stackrel{\text{def}}{=} \dot{\mathbf{x}}(t) = (\mathbf{x}^{(n+1)} - \mathbf{x}^{(n)})/\Delta t$ .
- 

The resulting ODE system is solved by the implicit Euler method, and for the linearized system of equations the iterative method Bi-CGSTAB (see [18]) is applied with implicit diagonal preconditioning.

For the convection-diffusion equation (10) it is appropriate to make use of an *implicit-explicit* time-integration method (see [1] for more details). The main advantage is that solving a nonlinear system, with for instance Newton's method, can be avoided, while still having reasonable stability properties, at least for mildly stiff equations. We will use within this class of integrators the first-order method 1-SBDF. Applied to (10), the discretization then reads:

$$(I - \epsilon_m \Delta t \bar{\Delta}) A_3^{(n+1)} = (I - \Delta t \mathbf{v}^{(n)} \cdot \bar{\nabla}) A_3^{(n)}, \quad (21)$$

where  $\bar{\Delta} A_3^{(n+1)}$  and  $\bar{\nabla} A_3^{(n)}$  are the semi-discretized approximations of the second-order derivative and the first-order derivative terms, respectively, in equation (10). A similar derivation can be done for the transformed PDE (14), in which the mesh velocities  $x_\theta$  and  $y_\theta$  play a role and are evaluated in the algorithm using values from the previous time level  $t_n$  as well. The non-symmetric linear system  $\mathcal{A}^{(n)} A_3^{(n+1)} = b^{(n)}$  behind (21) is again solved with the iterative method Bi-CGSTAB with implicit diagonal preconditioning. It is common practice to smooth the monitor function in moving mesh methods. This is because the computed monitor function is often very un-smooth. At the same time, a smoother monitor leads to a smoother mesh (a smoother transformation) and also makes the adaptive mesh PDEs easier to integrate

in time. Moreover, smoother meshes are also important when we investigate the local truncation errors on non-uniform meshes (see, e.g., [21]). In order to obtain smoother transitions in the mesh, rather than merely using equations (20), an additional filter (see also [16]) is applied to the weight functions. Instead of working with  $\omega_{ij}$ , the smoothed values

$$\bar{\omega}_{ij} \leftrightarrow \frac{1}{4}\omega_{ij} + \frac{1}{8}(\omega_{i+1,j} + \omega_{i-1,j} + \omega_{i,j+1} + \omega_{i,j-1}) + \frac{1}{16}(\omega_{i-1,j-1} + \omega_{i-1,j+1} + \omega_{i+1,j-1} + \omega_{i+1,j+1})$$

are being used in the mesh equations. This weighted sum corresponds with averaging the influence of the monitor values at neighbouring mesh points.

## 5 Numerical results

### 5.1 An implosion model

The first testmodel does not stem from MHD, but serves to show the capability of the proposed adaptive mesh approach to deal with extremely thin layers or peaks. The hyperbolic PDE

$$\frac{\partial A_3}{\partial t} - x \frac{\partial A_3}{\partial x} - y \frac{\partial A_3}{\partial y} = 0 \quad (22)$$

describes an ‘implosion’ of the initial solution  $A_3|_{t=0} = e^{-x^2-y^2}$  due to the velocity field  $\mathbf{v} = (-x, -y)^T$ . Starting with a Gaussian hump, the exact solution shrinks to a very sharp needle for increasing time, thereby keeping the maximum of the solution equal to 1 for all time. We solve the equation numerically on  $\Omega_p = [-3, 3] \times [-3, 3]$  for  $t \in [0, 8]$  on a  $60 \times 60$  mesh with timestep  $\Delta t = 10^{-3}$ . Numerical results at  $t = 8$  are shown in Figure 6. It should be obvious, when we observe the mesh (zoomed in several times around  $(0, 0)$ ) and the needle-type solution, that only a uniform mesh with *many* more mesh points might be able to resolve such small spatial scales. Surprisingly, the adaptive non-uniform mesh produces *stable* numerical solutions, although central differences in space and forward differences in time are being used. It seems that the adaptive moving mesh system in the decoupled numerical PDE procedure adds a small amount of diffusion to the

transformed hyperbolic PDE (22). But this is merely an experimental observation at this stage. This unexpected and interesting phenomenon deserves further theoretical analysis.

In the following three numerical experiments we follow different scenarios as proposed by Weiss [19]. In all the MHD testcases we use a  $70 \times 70$ , initially uniformly distributed, mesh and a timestep  $\Delta t = 10^{-3}$ .

## 5.2 Four-cell convection

The first MHD testcase is equation (10) with velocity field (11). The resistivity is chosen  $\epsilon_m = 5 \times 10^{-4}$ . In Figure 6 (top) the velocity field and the evolution of the parameter  $\beta$  as a function of time are displayed. Three phases can be recognized from the right plot: for small times the initially uniform magnetic field is distorted by the four convective cells to amplify the field. The magnetic field is dragged round by the motion and the field energy consequently rises. As the field grows, its scale of variation decreases until resistive effects become important. The flux is concentrated outside the eddies and so-called reconnection of the field lines takes place (second phase). In the final phase, the central field decays and a steady-state is reached. In Figure 6 we clearly see these phases from a different point of view as well. At times,  $t = 1/2, 1, 3$  and  $5$  the solution, the adaptive mesh and the magnetic field lines are depicted, respectively. The mesh is nicely concentrated in areas of high spatial activity, viz., the boundary and internal layers. Reconnection has taken place between  $t = 1$  and  $t = 3$ : the magnetic fieldlines have reconnected in regions of strong currents.

## 5.3 A single eddy

For this example we choose  $\epsilon_m = 10^{-3}$ ,  $\mathbf{v} = (32y(1 - 4y^2)^3 \cos(\pi x)/\pi, -(1 - 4y^2)^4 \sin(\pi x))^T$  and  $\Omega_p = [-0.5, 0.5] \times [-0.5, 0.5]$ . The velocity field and  $\beta(t)$  can be found in Figure 6 (middle plot). Figure 6 shows solutions, meshes and magnetic field lines at  $t = 1, 2, 3$  and  $10$ . We see the expulsion of flux from the single eddy. The flux is rapidly concentrated at the edges of the cell. The central field is amplified and finally decays. The process is completed at  $t = 10$  at which a steady-state is reached, where the adaptive mesh is located around the steep transitions of the solution.

## 5.4 Band of eddies in a horizontal field

In the third example the incompressible velocity field is taken to be  $\mathbf{v} = (8y(1 - 4y^2)^3 \sin(4\pi x)/\pi, \cos(4\pi x)(1 - 4y^2)^4)^T$  and the initial solution is rotated over  $90^\circ$ :  $A_3|_{t=0} = 1 - y$  with  $\Omega_p = [-0.5, 0.5] \times [-0.5, 0.5]$ . In Figures 6 (bottom) and 6 we display the numerical solutions and adaptive meshes at  $t = 1/4, 1/2, 1, 10$  for  $\epsilon_m = 10^{-3}$ . We clearly recognize the different phases again as in the previous two examples. The adaptive mesh method is capable to automatically track the regions of high gradients in the solution throughout the whole period of time. Note that steady-state has not yet completely been reached, but all main effects are visible, though, in this run.

## 6 Conclusions

In this paper we have applied an adaptive moving mesh technique to resistive 2D magnetic field evolution simulations. The adaptive method, based on a transformation between physical and computational coordinates, is derived from the minimization of a mesh-energy functional. Non-singularity of this transformation can be guaranteed for one- and two- space dimensions. Within the adaptive strategy a monitor function has been used with a *time-dependent*, automatically chosen parameter, in contrast to many previous papers in which the adaptivity parameter had to be adjusted for each new PDE model. Further study on the monitor functions, on the skewness of the mesh, and on the use of more sophisticated filters should be considered in the next step in the research. The recent paper [8] could be very helpful in this respect. Application of the current approach to full two-dimensional HD and MHD models (such as the ones described in [11]) is in progress. Note that the paper [16] may serve as a good starting point in this respect. In the latter type of models, the relation (6) needs special numerical care as well (for a set of solutions to this specific problem, see [5]). Other important issues are the application of the method to non-convex domains, for which we have to use the more complicated Euler-Lagrange equations coming from (15) rather than the elliptic system (17), and the 3D case, for which a proof of non-singularity of the transformation is not yet available (see [6] for more details).



## References

- [1] U. M. Ascher, S. J. Ruuth & B. T. R. Wetton, Implicit-explicit methods for time-dependent PDE's, *SIAM J. Num. Anal.* **32** (1995), 797-823.
- [2] G. Beckett, J. A. Mackenzie, A. Ramage & D. M. Sloan, On the numerical solution of one-dimensional PDEs using adaptive methods based on equidistribution, *J. of Comp. Phys.* **167** (2001), 372-392.
- [3] G. Beckett, J. A. Mackenzie, A. Ramage & D. M. Sloan, Computational solution of two-dimensional unsteady PDEs using moving mesh methods, *J. of Comp. Phys.*, **182** (2002), 478-495.
- [4] J. U. Brackbill, An adaptive grid with directional control, *J. of Comp. Phys.* **108** (1993), 38-50.
- [5] J. U. Brackbill & D. C. Barnes, The effect of nonzero  $\nabla \cdot \mathbf{B}$  on the numerical solution of the magnetohydrodynamic equations, *J. of Comp. Phys.* **35** (1980), 426-430.
- [6] Ph. Clément, R. Hagmeijer & G. Smeers, On the invertibility of mappings arising in 2D grid generation problems, *Num. Math.* **73** (1996), 37-51.
- [7] P. A. Davidson, *An introduction to magneto-hydrodynamics*, Cambridge Univ. Press, 2001.
- [8] W. Huang & W. Sun, Variational mesh adaptation II: error estimates and monitor functions, *J. of Comp. Phys.* **184** (2003), 619-648.
- [9] W. Huang & R. D. Russell, Moving mesh strategy based on a gradient flow equation for two-dimensional problems, *SIAM J. Sci. Comp.* **3** (1999), 998-1015.
- [10] W. Huang, L. Zheng & X. Zhan, Adaptive moving mesh methods for simulating one-dimensional groundwater problems with sharp moving fronts, *Int. J. for Num. Meth. in Eng.* **54** (2002), 1579-1603.
- [11] R. Keppens, G. Tóth, R. H. J. Westermann & J. P. Goedbloed, Growth and saturation of the Kelvin-Helmholtz instability with parallel and antiparallel magnetic fields, *J. Plasma Phys.* **61** (1999), 1-19.

- [12] R. Keppens, M. Nool, P. A. Zegeling & J. P. Goedbloed, Dynamic grid adaptation for computational magnetohydrodynamics Proc. High Performance Computing and Networking 2000, Eds. M. Bubak M. et al. (Springer-Verlag, Berlin Heidelberg), *Lecture Notes in Computer Science* **1823** (2000), 61-70.
- [13] P. Knupp & S. Steinberg, *Fundamentals of grid generation*, CRC-Press, Inc., 1993.
- [14] F. Liu, S. Ji & G. Liao, An adaptive grid method and its application to steady Euler flow calculations, *SIAM J. Sci. Comp.* **20** (1998), 811-825.
- [15] H. K. Moffat, *Magnetic field generation in electrically conducting fluids*, Cambridge Univ. Press, 1978,
- [16] H. Z. Tang & T. Tang, Adaptive mesh methods for one- and two-dimensional hyperbolic conservation laws, *SIAM J. Num. Anal.* **41** (2003), 487-515.
- [17] R. A. Trompert, Local uniform grid refinement for time-dependent partial differential equations, CWI-tract No. 107, Centrum voor Wiskunde en Informatica, Amsterdam, 1995.
- [18] H. A. van der Vorst, Bi-CGSTAB: A fast and smoothly converging variant of Bi-CG for the solution of nonsymmetric linear systems, *SIAM J. Sci. Stat. Comp.* **13** (1992), 631-644.
- [19] N. O. Weiss, The expulsion of magnetic flux by eddies, *Proc. of Roy. Soc. A* **293** (1966), 310-328.
- [20] A. M. Winslow, Adaptive mesh zoning by the equipotential method, *Technical Report UCID 19062* (1981).
- [21] P. A. Zegeling, Tensor-product adaptive grids based on coordinate transformations, *J. of Comp. Appl. Maths.* **166** (2004), 343-360.
- [22] P. A. Zegeling & R. Keppens, Adaptive MOL for magneto-hydrodynamic PDE models, In: Alain Vande Wouwer, Philippe Saucez, William E. Schiesser (eds.), *Adaptive Method of Lines*, Chapman & Hall/CRC Press (2001), 117-137.

- [23] P. A. Zegeling & H. P. Kok, Adaptive moving mesh computations for reaction-diffusion systems, *J. of Comp. Appl. Maths.* **168** (2004), 519-528.
- [24] U. Ziegler, A three-dimensional Cartesian adaptive grid code for compressible magnetohydrodynamics, *Comp. Phys. Comm.* **116** (1999), 65-77.

### **Acknowledgements**

The author would like to thank Rony Keppens (FOM-Institute, Nieuwegein) for the interesting discussions regarding the resistive MHD model, and also the referees for their valuable comments which have improved the quality of the paper significantly.

## Figure captions

### Figure 1:

Left panel: the structure of the asymptotic solution for  $\epsilon_m = 0$  and  $t \gtrsim 0$ .  
Right panel: the structure of the boundary and internal layers for  $0 < \epsilon_m \ll 1$  and  $t \rightarrow \infty$ .

### Figure 2:

The non-uniform mesh points seen as mass points connected by springs with springforces  $F$  representing values which are related to the monitor function  $\omega$ .

### Figure 3:

Left panel: the velocity field for the implosion model, middle panel: the adaptive mesh at  $t = 8$  (zoomed in around the origin), right panel: the resolved needle with the adaptive mesh.

### Figure 4:

Velocity fields and time-evolution of  $\beta(t)$  for each of the three MHD testcases (top: four-cell convection, middle: a single eddy, bottom: a band of eddies in a horizontal field).

### Figure 5:

Four-cell convection results at  $t = 1/2, 1, 3$  and  $5$  (left: solution, middle: mesh, right: magnetic field lines).

### Figure 6:

Single eddy results at  $t = 1, 2, 3$  and  $10$  (left: solution, middle: mesh, right: magnetic field lines).

**Figure 7:**

Band of eddies results at  $t = 1/4, 1/2, 1$  and  $10$  (left: solution, middle: mesh, right: magnetic field lines).

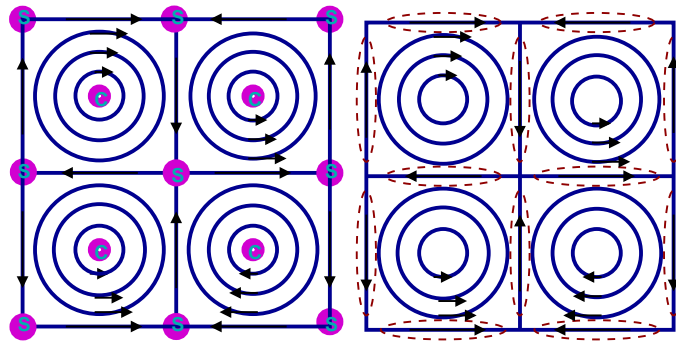


Figure 1:

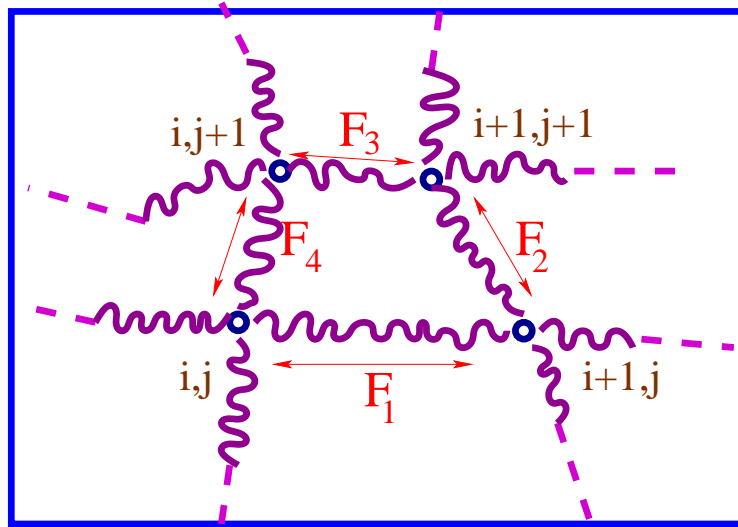


Figure 2:

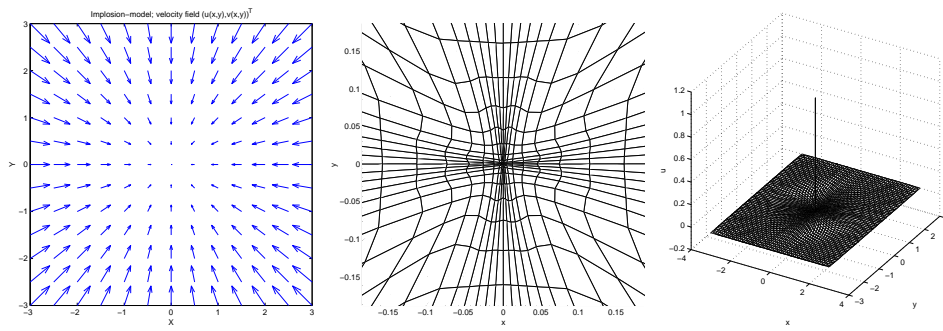


Figure 3:



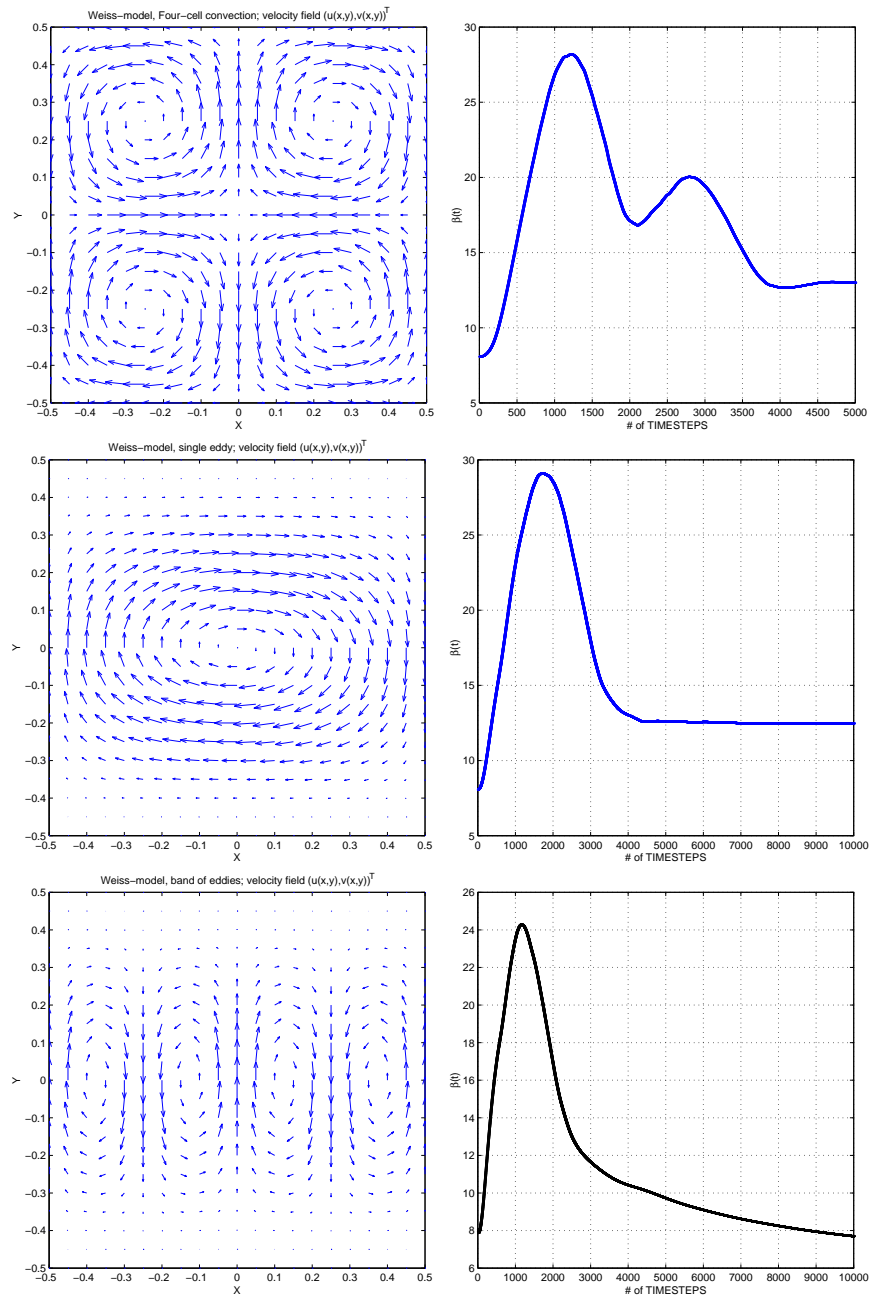


Figure 4:

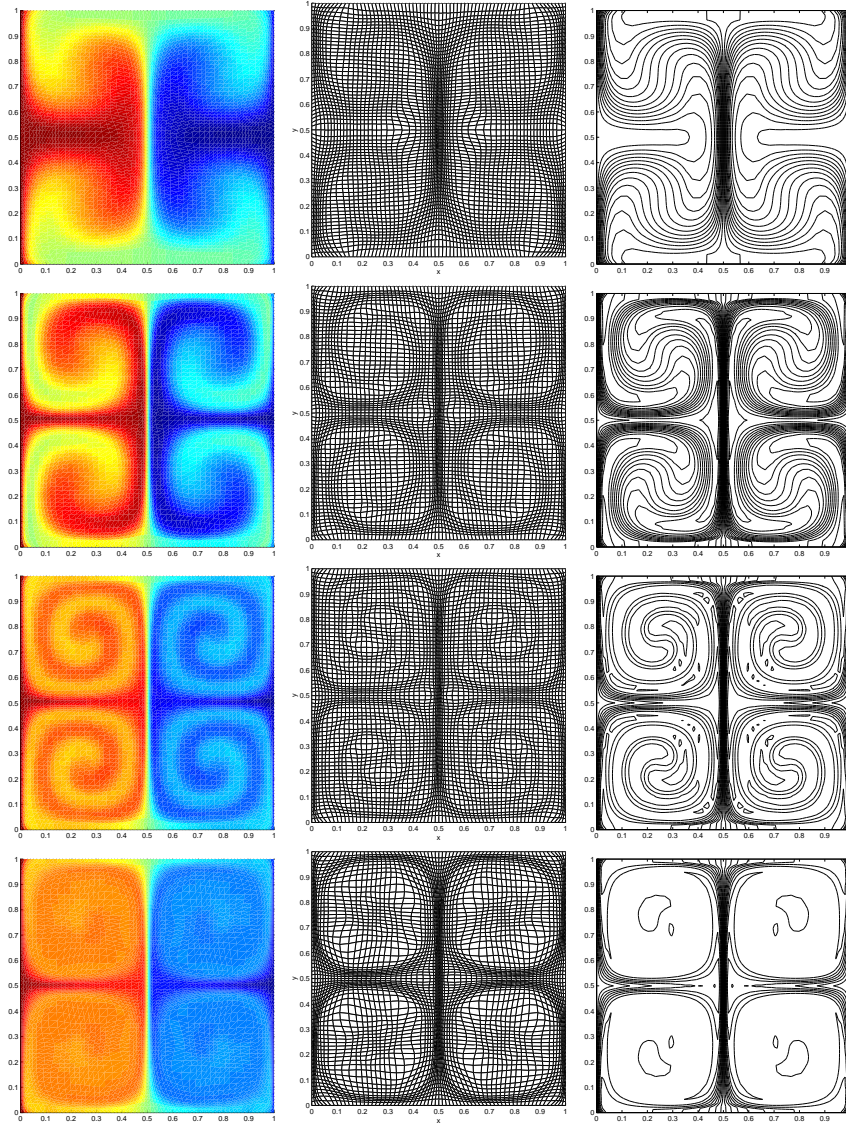


Figure 5:

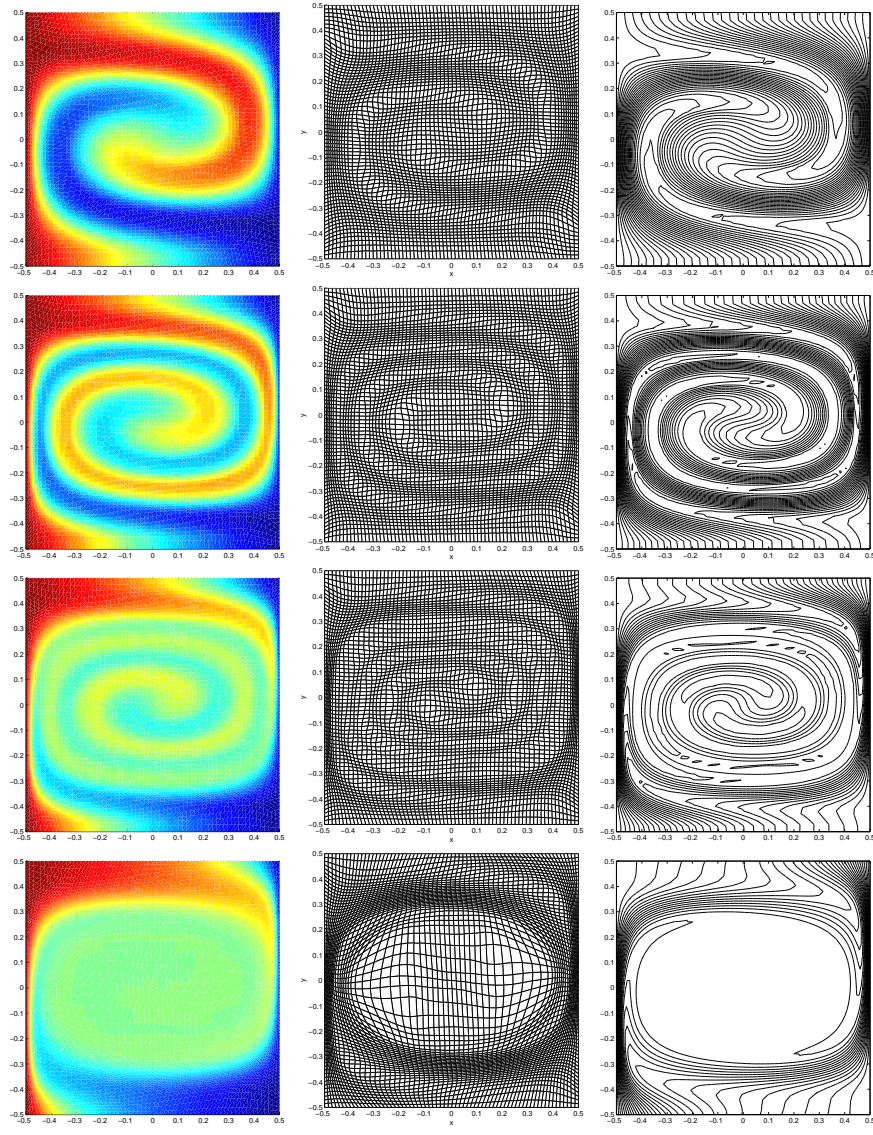


Figure 6:

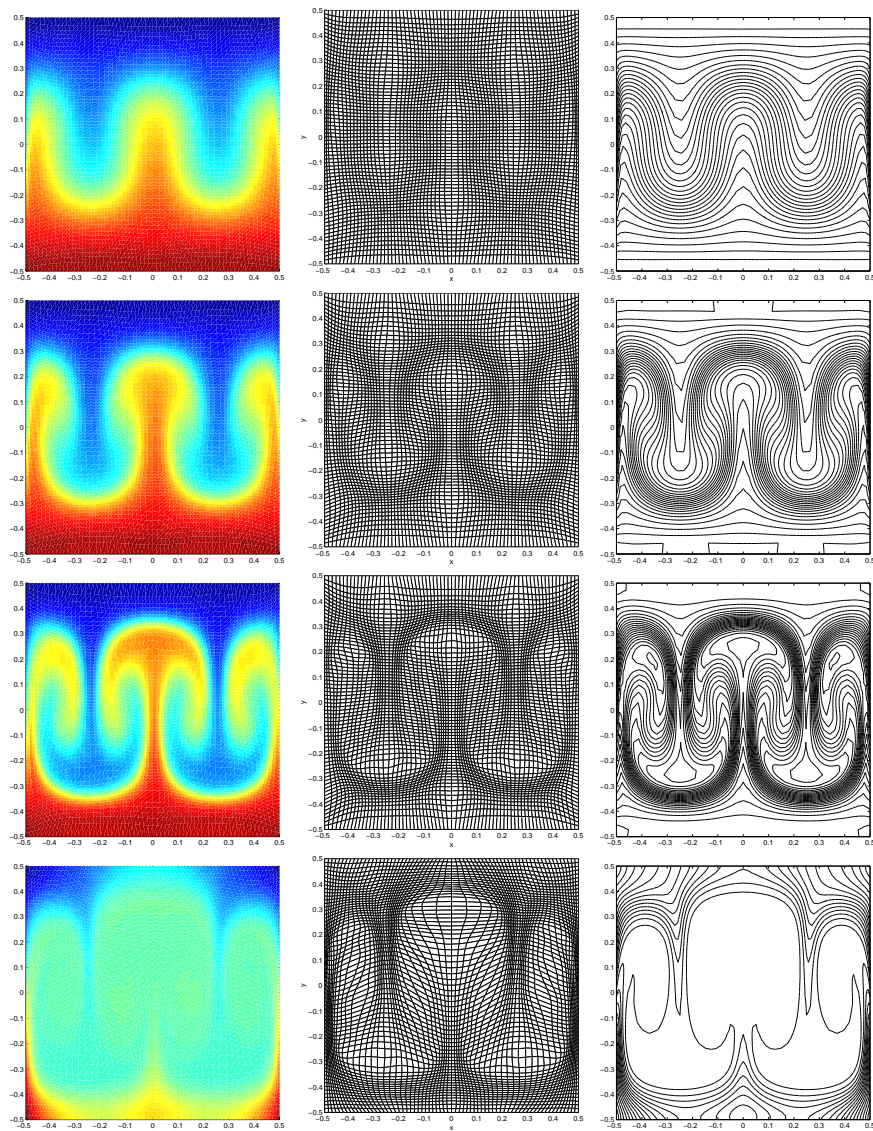


Figure 7: