

# Development of two-dimensional magneto-hydrodynamic simulation code in cylindrical geometry using the discontinuous Galerkin finite element method

K. Lee<sup>1</sup>, K.-J. Chung<sup>2</sup>, D.-K. Kim<sup>3</sup>, S.-G. Lee<sup>1</sup> & Y. S. Hwang<sup>1,2</sup>

<sup>1</sup>*Department of Nuclear Engineering, Seoul National University, Korea*

<sup>2</sup>*Center for Advance Research in Fusion Reactor Engineering, Seoul National University, Korea*

<sup>3</sup>*Agency for Defense Development, Korea*

## Abstract

Discontinuous Galerkin finite element method (DG-FEM) is applied for modelling magneto-hydrodynamics of electrically discharging plasma channel (DPC). For the exact application of a material model for DPC and surrounding media, the Lagrangian forms of equations are formulated in cylindrical geometry. This paper presents procedures of formulation and development of two-dimensional simulation code which includes prescribed features with details. To simplify algorithms, element-matrix relations are obtained analytically using linear triangular element and linear shape function. Numerical schemes and sequence of calculation are also discussed with a numerical result to demonstrate the feasibility of proposed method.

*Keywords:* discontinuous Galerkin finite element formulation, Lagrangian simulation, magneto-hydrodynamics, electrical discharge.

## 1 Introduction

In the early stage of electrical discharge process in water, exploding material quickly turns into the plasma state of high pressure, which is called non-ideal plasma. Analysis of non-ideal plasmas requires advanced modelling of material properties and accurate numerical simulation due to its unsteady behaviour, as one can observe in underwater electrical wire explosion or water gap arc



discharge [1]. Also, equation of state (EOS) and transport parameters such as thermal and electrical conductivities of non-ideal plasma play an important role in the numerical simulation [2]. Therefore, research groups on this subject had developed their own magneto-hydrodynamic (MHD) simulation code [3, 4] independently to accommodate their material model for non-ideal plasma. Arbitrary Eulerian-Lagrangian (ALE) or Lagrangian formulation was adopted in those codes for the effective description of unsteady non-ideal plasmas with surrounding media.

In terms of numerical method, techniques solving MHD equations had been extensively discussed in various fields of science and engineering. The finite difference method (FDM) is most classical approach in the field of computational fluid dynamics (CFD) due to its ease of implementation. Also, the finite volume method (FVM) became popular recently, since it inherently satisfies the conservation laws. However, both methods contain essential limitations in the treatment of complex geometries and inhomogeneous material with higher order accuracy. On the other hand, finite element method (FEM) is accepted as a more powerful tool in structural and electromagnetic (EM) analysis because it is easy to handle complicated systems with unstructured mesh [5]. In virtue of its generality came from Galerkin approximation, it is even applicable to MHD problems. However, the size of an assembled global matrix increases rapidly with the size of system. Solving a large size of matrix equation in each time step is too expensive.

In recent years, the Discontinuous Galerkin (DG) method has been developed and quite impressive progress has been made both fields of CFD [6, 7] and EM [8]. DG method introduces numerical fluxes instead of assembling process, allowing us to perform element-wise calculation. In addition to all the merits of standard FEM, it is free to choose basis functions and shape of elements, and easy to parallelize the algorithms [6]. Thus the combination of DG and FEM, so called DG-FEM, is a truly attractive method to solve unsteady MHD problem.

As a part of code development, this paper presents Lagrangian formulation of resistive MHD equations in axi-symmetric cylindrical geometry using DG-FEM. The detailed discretization process for the MHD equation sets with DG-FEM using simple shape of element and shape function is described in Sec. 2. In Sec. 3, procedure of code development and numerical scheme are discussed with some remarks on time advancement techniques. Validation of the code for simple shock tube problem is also described in Sec. 3.

## 2 Formulation

### 2.1 Governing equations

We consider inviscid, compressible unsteady flow for discharging plasma channel. If we also assume the plasma is a single fluid, the single fluid MHD equations can be organized. By rearranging and substitutions of momentum conservation, Maxwell equations, Ohm's law and energy conservation, following



set of MHD equations is derived in Lagrangian form. The meaning of notations and their dimensions are given in table 1.

$$\text{Equation of motion:} \quad \bar{u} = \frac{\partial \bar{r}}{\partial t}, \quad (1)$$

$$\text{Momentum equation:} \quad \rho \frac{D\bar{u}}{Dt} = -\bar{\nabla} p + \bar{J} \times \bar{B}, \quad (2)$$

$$\text{Induction equation:} \quad \frac{D\bar{B}}{Dt} = \eta \nabla^2 \bar{B} + (\bar{u} \cdot \bar{\nabla}) \bar{B}, \quad (3)$$

$$\text{Energy equation:} \quad \rho \frac{D\varepsilon}{Dt} = -p \bar{\nabla} \cdot \bar{u} + \bar{\nabla} \cdot (\kappa \bar{\nabla} T) + \frac{J^2}{\sigma} - Q_{rad}, \quad (4)$$

$$\text{Ampere's law:} \quad \bar{\nabla} \times \bar{B} = \mu \bar{J}, \quad (5)$$

$$\text{Equation of state:} \quad p = p(\rho, T), \quad \varepsilon = \varepsilon(\rho, T), \quad (6)$$

$$\text{Transport parameters:} \quad \kappa = \kappa(\rho, T), \quad \sigma = \sigma(\rho, T). \quad (7)$$

Table 1: List of symbols with dimension.

Notation	Meaning	Dimension
$\bar{r}$	Position	[m]
$\bar{u}$	Velocity	[m/s]
$\rho$	Mass density	[kg/m <sup>3</sup> ]
$\nu$	Specific volume	[m <sup>3</sup> /kg]
$p$	Pressure	[Pa]
$\bar{J}$	Electric current density	[A/m <sup>2</sup> ]
$\bar{B}$	Magnetic flux density	[T]
$\sigma$	Electrical conductivity	[S/m]
$\kappa$	Thermal conductivity	[W/m·K]
$\eta$	Magnetic diffusivity	[m <sup>2</sup> /s]
$\varepsilon$	Internal energy density	[J/kg]
$Q_{rad}$	Radiative power loss per unit volume	[W/m <sup>3</sup> ]

## 2.2 Discontinuous Galerkin finite element formulation

Azimuthal-symmetry ( $\partial/\partial\theta=0$ ) in cylindrical geometry is necessary to reduce those general equations to two dimensional equations. According to Ampere's law, there is only azimuthal component for magnetic flux density. Splitting second-order equations into two first-order differential equations allows us make a system of first-order partial differential equations. We introduced new notations  $\varepsilon_T = \partial\varepsilon/\partial T$ ,  $\varepsilon_\nu = \partial\varepsilon/\partial\nu$  to express energy equation in terms of temperature by chain rule, since most EOS models and material data are organized with respect to mass density ( $\rho$ ) and temperature ( $T$ ) [1].

Derivation of weak forms of eqns (2)–(4) with Galerkin FEM are given as eqns (8)–(15). Note that the eqns (10), (11), (13), (14) show the definition of the newly introduced variables  $q_T$  and  $q_B$ . Also,  $\Psi$  stands for a set of weighting functions which belongs to the same function space ( $V_h$ ) with the shape functions when Galerkin method is applied. Thus the subscript  $h$  means approximation with basis of  $V_h$ .

*Momentum equations:*

$$\int_{\Omega_e} \left( \rho \dot{u}_{r,h} + \frac{\partial p_h}{\partial r} + \frac{B_\theta}{\mu} \frac{\partial B_{\theta,h}}{\partial r} + \frac{B_\theta}{\mu r} B_{\theta,h} \right) \Psi r dr dz = 0, \quad (8)$$

$$\int_{\Omega_e} \left( \rho \dot{u}_{z,h} + \frac{\partial p_h}{\partial z} + \frac{B_\theta}{\mu} \frac{\partial B_{\theta,h}}{\partial z} \right) \Psi r dr dz = 0. \quad (9)$$

*Induction equations:*

$$\int_{\Omega_e} \left( q_{Br,h} - \frac{\partial B_{\theta,h}}{\partial r} \right) \Psi r dr dz = 0, \quad (10)$$

$$\int_{\Omega_e} \left( q_{Bz,h} - \frac{\partial B_{\theta,h}}{\partial z} \right) \Psi r dr dz = 0, \quad (11)$$

$$\int_{\Omega_e} \left( \dot{B}_{\theta,h} - \eta \left[ \frac{\partial q_{Br,h}}{\partial r} + \frac{\partial q_{Bz,h}}{\partial z} + \frac{q_{Br,h}}{r} - \frac{B_{\theta,h}}{r^2} \right] - [u_r q_{Br,h} + u_z q_{Bz,h}] \right) \Psi r dr dz = 0. \quad (12)$$

*Energy equations:*

$$\int_{\Omega_e} \left( q_{Tr,h} - \kappa \frac{\partial T_h}{\partial r} \right) \Psi r dr dz = 0, \quad (13)$$

$$\int_{\Omega_e} \left( q_{Tz,h} - \kappa \frac{\partial T_h}{\partial z} \right) \Psi r dr dz = 0, \quad (14)$$

$$\int_{\Omega_e} \left( \varepsilon_T \dot{T}_h + (p + \varepsilon_v) \dot{v} - v \left[ \bar{\nabla} \cdot (\bar{q}_T) + \frac{J^2}{\sigma} - Q_{rad} \right] \right) \Psi r dr dz = 0. \quad (15)$$

In order to make clear and simple formulation using DG-FEM, we chose linear triangular element and linear shape function. Feature of arbitrary element is shown in fig. 1. DG numerical fluxes [9] were adopted to treat boundary line integration of element. Notations for jump  $[\ ]$  and mean value  $\{ \}$  are also defined to specify the numerical fluxes. Variables  $\phi$  and  $\bar{q}$  stand for general variable and its gradient, respectively.

*Jumps and Mean values:*

$$\begin{aligned} \{ \phi \} &= \frac{1}{2} (\phi + \phi_{NB}), \quad \{ \bar{q} \} = \frac{1}{2} (\bar{q} + \bar{q}_{NB}), \\ [ \phi ] &= (\phi - \phi_{NB}) \cdot \hat{n}, \quad [ \bar{q} ] = (\bar{q} - \bar{q}_{NB}) \cdot \hat{n}, \end{aligned} \quad (16)$$

Numerical fluxes (DG model):

$$\begin{aligned}\hat{\phi}_h &= \{\phi_h\} + \mathbf{C}_{12}[\phi_h] - C_{22}[\bar{q}_h], \\ \hat{\bar{q}}_h &= \{\bar{q}_h\} - C_{11}[\phi_h] - \mathbf{C}_{12}[\bar{q}_h].\end{aligned}\quad (17)$$

Now, above equations can be discretized in space by applying the Gauss-divergence theorem to treat the area integral over the element, as we do in usual finite element formulation [11]. In order to simplify the notations, all the terms are categorized into seven types. Please refer to the Appendix for the definition of notations. The subscript  $l$  denotes the coordinates, i.e.  $l=r, z$ .

Type I:

$$\frac{\partial \phi_h}{\partial r} \rightarrow \sum_{s=1}^3 \mathbf{D}_{r,s} \phi + \sum_{s=1}^3 \mathbf{A}_{r,s} \phi_{NB,s} - \sum_{s=1}^3 \mathbf{J}_{rr,s} q_r + \sum_{s=1}^3 \mathbf{J}_{rz,s} q_z - \mathbf{H}_r \phi, \quad (18)$$

Type II:

$$\frac{\partial \phi_h}{\partial z} \rightarrow \sum_{s=1}^3 \mathbf{D}_{z,s} \phi + \sum_{s=1}^3 \mathbf{A}_{z,s} \phi_{NB,s} - \sum_{s=1}^3 \mathbf{J}_{rz,s} q_r + \sum_{s=1}^3 \mathbf{J}_{zz,s} q_z - \mathbf{H}_z \phi, \quad (19)$$

Type III:

$$\frac{\partial q_{hl}}{\partial r} \rightarrow \sum_{s=1}^3 \mathbf{A}_{l,s} q_l + \sum_{s=1}^3 \mathbf{D}_{l,s} q_{l,NB,s} - \sum_{s=1}^3 \mathbf{G}_{ll,s} \phi + \sum_{s=1}^3 \mathbf{G}_{ll,s} \phi_{NB,s} - \mathbf{H}_r q_l, \quad (20)$$

Type IV:

$$\frac{\partial q_{hl}}{\partial z} \rightarrow \sum_{s=1}^3 \mathbf{A}_{l,s} q_l + \sum_{s=1}^3 \mathbf{D}_{l,s} q_{l,NB,s} - \sum_{s=1}^3 \mathbf{G}_{ll,s} \phi + \sum_{s=1}^3 \mathbf{G}_{ll,s} \phi_{NB,s} - \mathbf{H}_z q_l, \quad (21)$$

Type V:

$$\begin{aligned}\bar{\nabla} \cdot \bar{q} &\rightarrow \sum_{s=1}^3 \mathbf{A}_{r,s} q_r + \sum_{s=1}^3 \mathbf{A}_{z,s} q_z + \sum_{s=1}^3 \mathbf{D}_{r,s} q_{r,NB,s} + \sum_{s=1}^3 \mathbf{D}_{z,s} q_{z,NB,s} \\ &\quad - \sum_{s=1}^3 \mathbf{Q}_s \phi + \sum_{s=1}^3 \mathbf{Q}_s \phi_{NB,s} - \mathbf{H}_r q_r - \mathbf{H}_z q_z,\end{aligned}\quad (22)$$

$$\text{Type VI:} \quad \phi_h, q_{lh} \rightarrow r_m \mathbf{E} \phi \text{ and } r_m \mathbf{E} q_l, \quad (23)$$

$$\text{Type VII:} \quad c(\text{const.}) \rightarrow cr_m Y. \quad (24)$$

Substituting expressions (18)–(24) into each terms appear in eqns (8)–(15) leads us to obtain the element-matrix relations for each equation. Note that the non-linear terms related to magnetic flux density have been linearized. Also, mean radius approximation is taken in the integrands of eqns (8)–(15) to avoid singularity [12] at the axis. Finally, the resultant element-matrix relations are obtained in eqns. (25)–(32).

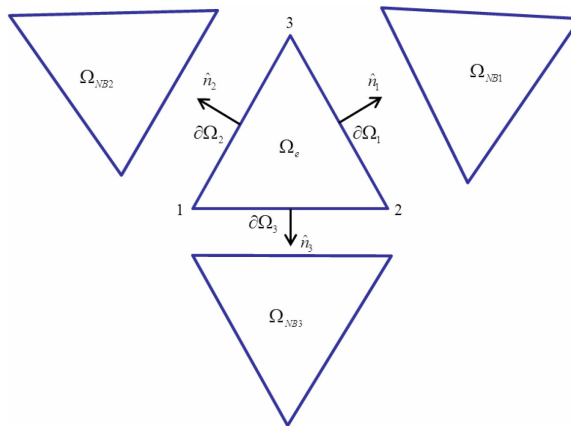


Figure 1: Sketch of an element and its neighbors.

*Momentum equations:*

$$(\mathbf{E}\dot{\underline{u}}_r) = -\frac{\nu}{r_m} \left( \sum_{s=1}^3 \mathbf{N}_{r,s} \underline{p} + \sum_{s=1}^3 \mathbf{N}_{r,s} \underline{p}_{NB,s} - \mathbf{H}_r \underline{p} \right) - \frac{\nu B_m}{\mu r_m} \left( \sum_{s=1}^3 \mathbf{D}_{r,s} \underline{B} + \sum_{s=1}^3 \mathbf{A}_{r,s} \underline{B}_{NB,s} - \sum_{s=1}^3 \mathbf{J}_{rr,s} \underline{q}_{Br} + \sum_{s=1}^3 \mathbf{J}_{rz,s} \underline{q}_{Bz} - \mathbf{H}_r \underline{B} \right), \quad (25)$$

$$(\mathbf{E}\dot{\underline{u}}_z) = -\frac{\nu}{r_m} \left( \sum_{s=1}^3 \mathbf{N}_{z,s} \underline{p} + \sum_{s=1}^3 \mathbf{N}_{z,s} \underline{p}_{NB,s} - \mathbf{H}_z \underline{p} \right) - \frac{\nu B_m}{\mu r_m} \left( \sum_{s=1}^3 \mathbf{D}_{z,s} \underline{B} + \sum_{s=1}^3 \mathbf{A}_{z,s} \underline{B}_{NB,s} - \sum_{s=1}^3 \mathbf{J}_{rz,s} \underline{q}_{Br} + \sum_{s=1}^3 \mathbf{J}_{zz,s} \underline{q}_{Bz} - \mathbf{H}_z \underline{B} \right), \quad (26)$$

*Induction equations:*

$$(\mathbf{E}\underline{q}_{Br}) = \frac{1}{r_m} \left( \sum_{s=1}^3 \mathbf{D}_{r,s} \underline{B} + \sum_{s=1}^3 \mathbf{A}_{r,s} \underline{B}_{NB,s} - \sum_{s=1}^3 \mathbf{J}_{rr,s} \underline{q}_{Br} + \sum_{s=1}^3 \mathbf{J}_{rz,s} \underline{q}_{Bz} - \mathbf{H}_r \underline{B} \right), \quad (27)$$

$$(\mathbf{E}\underline{q}_{Bz}) = \frac{1}{r_m} \left( \sum_{s=1}^3 \mathbf{D}_{z,s} \underline{B} + \sum_{s=1}^3 \mathbf{A}_{z,s} \underline{B}_{NB,s} - \sum_{s=1}^3 \mathbf{J}_{rz,s} \underline{q}_{Br} + \sum_{s=1}^3 \mathbf{J}_{zz,s} \underline{q}_{Bz} - \mathbf{H}_z \underline{B} \right), \quad (28)$$

$$(\mathbf{E}\dot{\underline{B}}) = \frac{\eta}{r_m} \left( \sum_{s=1}^3 \mathbf{A}_{r,s} \underline{q}_{Br} + \sum_{s=1}^3 \mathbf{D}_{r,s} \underline{q}_{Br,NB,s} - \sum_{s=1}^3 \mathbf{G}_{rr,s} \underline{B} + \sum_{s=1}^3 \mathbf{G}_{rz,s} \underline{B}_{NB,s} - \mathbf{H}_r \underline{q}_{Br} + \sum_{s=1}^3 \mathbf{A}_{z,s} \underline{q}_{Bz} + \sum_{s=1}^3 \mathbf{D}_{z,s} \underline{q}_{Bz,NB,s} - \sum_{s=1}^3 \mathbf{G}_{zz,s} \underline{B} + \sum_{s=1}^3 \mathbf{G}_{zz,s} \underline{B}_{NB,s} - \mathbf{H}_z \underline{q}_{Bz} \right) + \mathbf{E} \left( \frac{\eta}{r_m} \underline{q}_{Br} - \frac{\eta}{r_m^2} \underline{B} + u_{rm} \underline{q}_{Br} + u_{zm} \underline{q}_{Bz} \right), \quad (29)$$

*Energy equations:*

$$\left( \mathbf{E} + \frac{1}{r_m} \sum_{s=1}^3 \mathbf{J}_{rr,s} \right) \underline{q}_{Tr} = \frac{1}{r_m} \left( \sum_{s=1}^3 \mathbf{D}_{r,s} \kappa T + \sum_{s=1}^3 \mathbf{A}_{r,s} \kappa T_{NB,s} + \sum_{s=1}^3 \mathbf{J}_{rz,s} \underline{q}_{Tz} - \mathbf{H}_r \kappa T \right), \quad (30)$$

$$\left( \mathbf{E} - \frac{1}{r_m} \sum_{s=1}^3 \mathbf{J}_{zs,s} \right) \underline{q}_{Tz} = \frac{1}{r_m} \left( \sum_{s=1}^3 \mathbf{D}_{zs,s} \kappa \underline{T} + \sum_{s=1}^3 \mathbf{A}_{zs,s} \kappa \underline{T}_{NB,s} - \sum_{s=1}^3 \mathbf{J}_{rz,s} \underline{q}_{Tr} - \mathbf{H}_z \kappa \underline{T} \right), \quad (31)$$

$$\begin{aligned} \varepsilon_T(\underline{\mathbf{E}}\dot{\mathbf{T}}) = & -\dot{\mathbf{v}}(\underline{\mathbf{E}}\dot{\mathbf{p}}) - \left( \dot{\mathbf{v}}\varepsilon_v - \nu \frac{J^2}{\sigma} + \nu Q_{rad} \right) (Y) + \frac{\nu}{r_m} \left( \sum_{s=1}^3 \mathbf{A}_{rs,s} \underline{q}_r + \sum_{s=1}^3 \mathbf{A}_{zs,s} \underline{q}_z \right. \\ & \left. + \sum_{s=1}^3 \mathbf{D}_{rs,s} \underline{q}_{Tr,NB,s} + \sum_{s=1}^3 \mathbf{D}_{zs,s} \underline{q}_{zr,NB,s} - \sum_{s=1}^3 \mathbf{Q}_s \kappa \underline{T} + \sum_{s=1}^3 \mathbf{Q}_s \kappa \underline{T}_{NB,s} - \mathbf{H}_r \underline{q}_r - \mathbf{H}_z \underline{q}_z \right). \end{aligned} \quad (32)$$

### 3 Numerical schemes and code development

#### 3.1 Numerical schemes

Eqns (26), (29) and (32) require time integration for a numerical solution of transient problem. Explicit schemes are usually preferred such as second- or higher-order Runge-Kutta methods (RK2, RK3,...), since they enhance the advantages of DG-FEM formulation. As a beginning phase of development, we adopted Explicit Euler (EE) method for the velocity and temperature, and RK2 for the position. Temporal accuracy can be improved later when the feasibility and suitability of formulation are acquired.

The induction equation is basically classified into magnetic diffusion equation. Unfortunately, magnetic diffusion time is seriously short for some cases such as electrical wire explosion in laboratory ( $\sim 10^{-16}$  sec). This time scale is much smaller than that of momentum and energy equations. Hence, using common time step satisfying CFL condition for induction equation is too weary to solve entire governing equations simultaneously. Therefore, we chose to solve equations one by one at a time.

In fact, various techniques to solve induction equation based on FEM had been reported. Thus, we can try one of those methods very flexibly, depending on the physical features of the problem. For example, magnetic diffusion in Z-pinch physics is handled with standard FEM formulation with implicit time stepping scheme [3] due to its large electric current ( $\sim$ MA) and complicated system. On the other hand, explicit scheme “cross” with FVM formulation has been used to solve all the MHD equations in the same time step for the one-dimensional UEWE analysis [4]. Also, the Crank-Nicolson time stepping method could be used with DG-FEM formulation [13] by dividing sub-regions.

From now on, we will drop the further discussions related to solving induction equations and will treat them in a future work. Note that even if we dropped the induction equations from the set of governing equations, we can still handle the problems such as simple thermodynamic problem such as cylindrical shock tube problem or MHD phenomena with mild current ( $\sim$ kA) such as UEWE and water gap arc discharges. Now the models for EOS and transport coefficients became much more important than the calculation of magnetic fields, for the problems just mentioned.

### 3.2 Computational mesh

As a part of pre-process, linear triangular area mesh is created using *Gmsh 2.6.1* [14]. Although this program is freeware, it provides four modules: geometry, mesh, solver and post-processing. But we utilized it only for generating the mesh file.

The drawback of *Gmsh* is that we cannot control the exact number of elements. Only the length of sides of edge element can be specified instead. However, as of our current status, this does not cause any serious problem.

### 3.3 Code and validation

Code has been written in C language. As already mentioned, all equations are solved one by one, based on the state of previous time step (see fig. 2.). First, the code reads mesh file and finds out the total number of elements and nodes. Then, required memory can be allocated to node and element data structures. After that, the code initialises variables. Geometrical data are also calculated such as surface outward normal vectors, length of sides, coordinates of center of mass, and the area of element from the data of mesh file. Physical data for the time, subzones, nodes and elements also can be initialized depending on the problem and material. Since we are using the linear triangular element, Jacobian and all the derivatives can be calculated analytically. Hence there is no need to use Gaussian quadrature for the numerical integration. This fact makes the code much nice and fast. After initialization, solver begins element-wise calculations for momentum equations and energy equations by direct inversion of element matrix relations. Now, thermodynamic properties can be obtained from EOS model. Thermal and

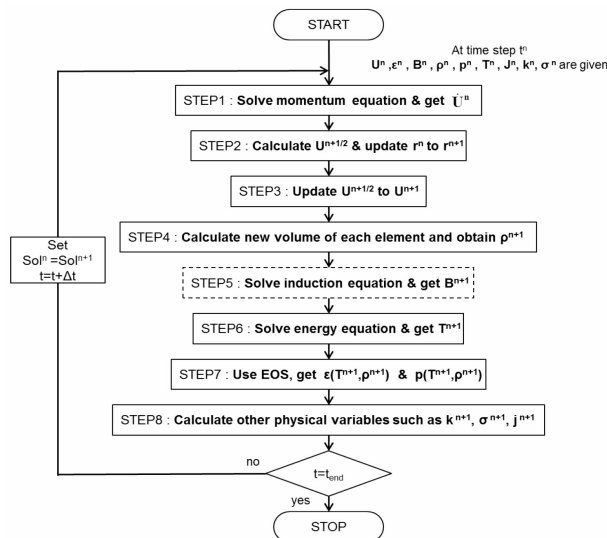


Figure 2: Flowchart of calculation.



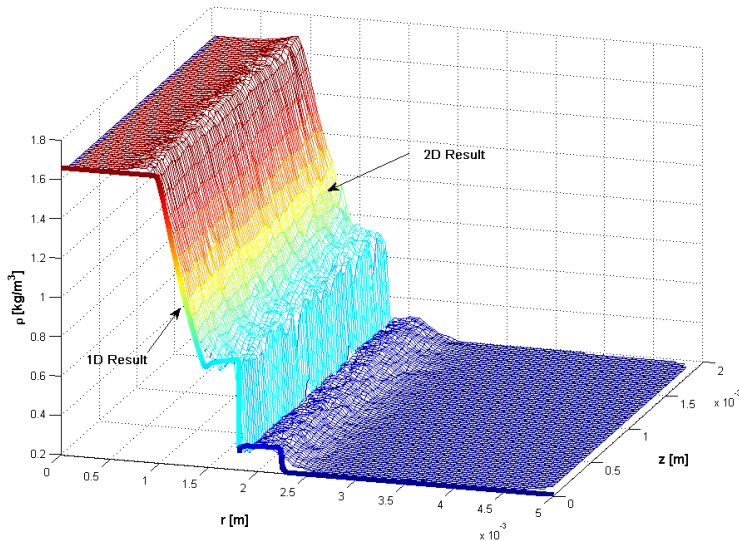


Figure 3: Numerical result of cylindrical shock tube problem ( $t=1.6 \mu\text{s}$ ).

electrical conductivities are also calculated from their physics models. The code will repeat the time advancement based on the data of current time step.

As a part of validation, crude numerical results for the cylindrical shock tube problem are presented in Fig. 3. The test gas is Argon and ideal EOS is applied. Both sides are initially in thermal equilibrium and the diaphragm was located at  $r=1.5\text{mm}$ . As we can see, results from 1D and 2D code agree to each other very well. However, there exists a bump behind the rarefaction wave, which is originated from numerical error. It should be eliminated by optimisation of code.

More strict validation can be made by comparing with the experimental results such as UEWE or water gap arc discharges with various EOS and transport parameter models [1, 15]. It should be the next work after this one.

## 4 Conclusion

The DG-FEM formulation of the resistive MHD equations and the development of simulation code have been described. The noticeable features of our code are: (i) to solve Lagrangian form of equations in cylindrical geometry, (ii) to handle inhomogeneous materials, and (iii) to accommodate independent EOS and transport parameter models. Although this is not the first trial to solving MHD equations with DG-FEM [16–18], the distinct procedure for the numerical implementation with those unique features has been established in this work. Further studies such as rigorous validation of the code with experimental results and advanced techniques solving induction equation will be handled in the future work.

## Acknowledgements

This work was supported by the Defense Research Laboratory Program of the Defense Acquisition Program Administration and the Agency for Defense Development of Republic of Korea.

## Appendix

$$\underline{Y} = \int_{\Omega_e} \Psi \, dr \, dz = \frac{S_e}{3} \begin{bmatrix} 1 & 1 & 1 \end{bmatrix}^T, \quad \mathbf{E} = \int_{\Omega_e} \Psi \Psi^T \, dr \, dz = \frac{S_e}{12} \begin{bmatrix} 2 & 1 & 1 \\ 1 & 2 & 1 \\ 1 & 1 & 2 \end{bmatrix},$$

where  $S_e = \frac{1}{2} |r_{31} \cdot z_{21} - r_{21} \cdot z_{31}|$ ,  $(z_{ij} = z_i - z_j, r_{ij} = r_i - r_j)$

$$\mathbf{H}_r = \int_{\Omega_r} r_m \frac{\partial \Psi}{\partial r} \Psi^T \, dr \, dz = \frac{r_m}{6} \begin{bmatrix} z_{23} & z_{23} & z_{23} \\ z_{31} & z_{31} & z_{31} \\ z_{12} & z_{12} & z_{12} \end{bmatrix}, \quad \mathbf{H}_z = \int_{\Omega_r} r_m \frac{\partial \Psi}{\partial z} \Psi^T \, dr \, dz = \frac{r_m}{6} \begin{bmatrix} r_{32} & r_{32} & r_{32} \\ r_{13} & r_{13} & r_{13} \\ r_{21} & r_{21} & r_{21} \end{bmatrix},$$

$$\mathbf{S}_s = \int_{\partial\Omega_s} \Psi \Psi^T \, ds, \quad (s=1,2,3; \text{"side index"});$$

$$\mathbf{S}_1 = \frac{L_1}{6} \begin{bmatrix} 0 & 0 & 0 \\ 0 & 2 & 1 \\ 0 & 1 & 2 \end{bmatrix}, \quad \mathbf{S}_2 = \frac{L_2}{6} \begin{bmatrix} 2 & 0 & 1 \\ 0 & 0 & 0 \\ 1 & 0 & 2 \end{bmatrix}, \quad \mathbf{S}_3 = \frac{L_3}{6} \begin{bmatrix} 2 & 1 & 0 \\ 1 & 2 & 0 \\ 0 & 0 & 0 \end{bmatrix},$$

$$\mathbf{N}_{r,s} = \left(\frac{1}{2}\right) r_m n_r \int_{\partial\Omega_s} \Psi \Psi^T \, ds = \left(\frac{1}{2}\right) r_m n_{r,s} \mathbf{S}_s,$$

$$\mathbf{N}_{z,s} = \left(\frac{1}{2}\right) r_m n_z \int_{\partial\Omega_s} \Psi \Psi^T \, ds = \left(\frac{1}{2}\right) r_m n_{z,s} \mathbf{S}_s,$$

$$\mathbf{D}_{r,s} = \left(\frac{1}{2} + C_{12}\right) r_m n_r \int_{\partial\Omega_s} \Psi \Psi^T \, ds = \left(\frac{1}{2} + C_{12}\right) r_m n_{r,s} \mathbf{S}_s,$$

$$\mathbf{D}_{z,s} = \left(\frac{1}{2} + C_{12}\right) r_m n_z \int_{\partial\Omega_s} \Psi \Psi^T \, ds = \left(\frac{1}{2} + C_{12}\right) r_m n_{z,s} \mathbf{S}_s,$$

$$\mathbf{A}_{r,s} = \left(\frac{1}{2} - C_{12}\right) r_m n_r \int_{\partial\Omega_s} \Psi \Psi^T \, ds = \left(\frac{1}{2} - C_{12}\right) r_m n_{r,s} \mathbf{S}_s,$$

$$\mathbf{A}_{z,s} = \left(\frac{1}{2} - C_{12}\right) r_m n_z \int_{\partial\Omega_s} \Psi \Psi^T \, ds = \left(\frac{1}{2} - C_{12}\right) r_m n_{z,s} \mathbf{S}_s,$$

$$\mathbf{J}_{rr,s} = (C_{22} r_m n_r n_r) \int_{\partial\Omega_s} \Psi \Psi^T \, ds = (C_{22} r_m n_{r,s} n_{r,s}) \mathbf{S}_s,$$

$$\mathbf{J}_{rz,s} = (C_{22} r_m n_r n_z) \int_{\partial\Omega_s} \Psi \Psi^T \, ds = (C_{22} r_m n_{r,s} n_{z,s}) \mathbf{S}_s,$$

$$\mathbf{J}_{zz,s} = (C_{22} r_m n_z n_z) \int_{\partial\Omega_s} \Psi \Psi^T \, ds = (C_{22} r_m n_{z,s} n_{z,s}) \mathbf{S}_s,$$

$$\mathbf{G}_{rr,s} = (C_{11} r_m n_r n_r) \int_{\partial\Omega_s} \Psi \Psi^T \, ds = (C_{11} r_m n_{r,s} n_{r,s}) \mathbf{S}_s,$$



$$\mathbf{G}_{z,s} = (C_{11}r_m n_z n_z) \int_{\partial\Omega_s} \Psi \Psi^T ds = (C_{11}r_m n_z n_z) \mathbf{S}_s ,$$

$$\mathbf{Q}_s = (C_{11}r_m) \int_{\partial\Omega_s} \Psi \Psi^T ds = (C_{11}r_m) \mathbf{S}_s .$$

## References

- [1] Chung, K. J. and Hwang, Y. S., Thermodynamic Properties and Electrical Conductivity of Water Plasma, *Contrib. Plasma Phys. (to be published)*, vol. 53, 2013.
- [2] Sheftman, D. and Krasik, Y. E., Evaluation of electrical conductivity and equations of state of non-ideal plasma through microsecond timescale underwater electrical wire explosion, *Physics of Plasmas*, vol. 18, pp. 092704-8, 2011.
- [3] Garasi, C. J., Bliss, D. E., Mehlhorn, T. A., Oliver, B. V., Robinson, A. C., and Sarkisov, G. S., Multi-dimensional high energy density physics modeling and simulation of wire array Z-pinch physics, *Physics of Plasmas*, vol. 11, pp. 2729-2737, 2004.
- [4] Grinenko, A., Gurovich, V. T., Saypin, A., Efimov, S., Krasik, Y. E., and Oreshkin, V. I., Strongly coupled copper plasma generated by underwater electrical wire explosion, *Physical Review E*, vol. 72, p. 066401, 2005.
- [5] Jin, J. M., *The finite element method in electromagnetics*, 2nd ed., Wiley: New York, 2002.
- [6] Li, B. Q., *Discontinuous Finite Elements in Fluid Dynamics and Heat Transfer*, Springer-Verlag: Germany, 2006.
- [7] Cockburn, B., Discontinuous Galerkin methods, *ZAMM - Journal of Applied Mathematics and Mechanics*, vol. 83, pp. 731-754, 2003.
- [8] Gedney, S., Chong, D. L., Guernsey, B., Roden, J. A., Crawford, R., and Miller, J. A., The Discontinuous Galerkin Finite Element Time Domain Method (DGFETD): A High Order, Globally-Explicit Method for Parallel Computation, in *EMC 2007. IEEE International Symposium on Electromagnetic Compatibility*, 2007., 2007, pp. 1-3.
- [9] Kirby, R. and Karniadakis, G., Selecting the Numerical Flux in Discontinuous Galerkin Methods for Diffusion Problems, *Journal of Scientific Computing*, vol. 22-23, pp. 385-411, 2005.
- [10] Castillo, P., Cockburn, B., Perugia, I., and Schötzau, D., An A Priori Error Analysis of the Local Discontinuous Galerkin Method for Elliptic Problems, *SIAM Journal on Numerical Analysis*, vol. 38, pp. 1676-1706, 2000.
- [11] Lyon, P. E., Axisymmetric finite element modeling for the design and analysis of cylindrical adhesive joints based on dimensional stability, Master of Science, Mechanical and Aerospace Engineering, UTAH STATE UNIVERSITY, Logan, Utah, 2011.
- [12] Clayton, J. D. and Rencis, J. J., Numerical integration in the axisymmetric finite element formulation, *Adv. Eng. Softw.*, vol. 31, pp. 137-141, 2000.

- [13] Smolentsev, S., Morley, N., and Abdou, M., Code development for analysis of MHD pressure drop reduction in a liquid metal blanket using insulation technique based on a fully developed flow model, *Fusion Engineering and Design*, vol. 73, pp. 83-93, 2005.
- [14] Geuzaine, C. and Remacle, J.-F., <http://geuz.org/gmsh/>
- [15] Kim, I., Kim, D.-K., Baek, S.-H., and Song, S.-Y., Magnetohydrodynamic Behaviour of Warm Dense Plasmas Created by Underwater Wire Explosion in *33rd EPS Conference on Plasma Phys.*, Rome, 19 - 23 June 2006, 2006, p. P5.064.
- [16] Warburton, T. C. and Karniadakis, G. E., A Discontinuous Galerkin Method for the Viscous MHD Equations, *Journal of Computational Physics*, vol. 152, pp. 608-641, 7/1/ 1999.
- [17] Halashi, B. K., Luo, H., Spicer, D., and MacNiece, P., A Discontinuous Galerkin Method for the Magnetohydrodynamics on Arbitrary Grids, in *50th AIAA Aerospace Sciences Meeting including the New Horizons Forum and Aerospace Exposition, 09 - 12 January 2012*, Nashville, Tennessee, 2012.
- [18] Houston, P., Schoetzau, D., and Wei, X., A Mixed Discontinuous Galerkin Method for Incompressible Magnetohydrodynamics, *Journal of Scientific Computing*, 2008.

