Derivation of jump conditions in multiphase incompressible flows with singular forces

Caesar O. Harahap[1)] and Hideaki Miura[2)]

The Graduate University for Advanced Studies (SOKENDAI), Toki 509-5292, Japan

[2)]*National Institute for Fusion Science, Toki 509-5292, Japan*

(Received 15 October 2007)

With the aim to numerically solve problems of plasma dynamics consisting of multiple phases and/or different governing equations, an immersed interface technique is extended to solve a system with mass density and viscosity jumps. The jump conditions for velocity, pressure and their derivatives (sets of coupled equations) are derived.

Keywords: dicontinuities, singular forces, jump conditions, immersed interface method

DOI: 10.1585/pfr.1.001

1 Introduction

Clarifying key physics of complex behaviors of a hot plasma in a magnetic confinement device such as the Large Helical Device (LHD) is essential for understanding experimental results. For this purpose, various kinds of simulation studies such as magnetohydrodynamics (MHD), two-fluid equations, Vlasov and gyrokinetic simulations have been conducted. (See, for example, Refs.[1, 2, 3]) The MHD simulation class is the simplest among them in the sense that the system of equations consists of a relatively few number of equations, and the number of dimensions of the independent variable space is only three. Nevertheless, there still remain many difficulties mimicking the device geometry in detail and taking various experimental and engineering parts into account. One of those difficulties is in connecting the hot plasma region and the vacuum region, which is complicated because the governing equations of these regions can be different from each other.

In MHD simulations, the vacuum region is often described by the MHD equations with very low pressure and/or mass density, or simply omitted from the simulation by imposing the boundary condition on the outermost magnetic surface. For the purpose of studying the effects of plasma deformations around the hot plasma boundary or to study peripheral regions and hot plasma core simultaneously, the former approach is preferable. In the former simulations, very large resistivity and/or viscosity are imposed to the low pressure region. However, jumps of physical variables such as mass density, pressure, and temperature often causes numerical oscillations. Although such oscillations may be avoided by adopting some numerical techniques such as the Godunov, TVD or CIP scheme, the computation becomes complex and the numerical accuracy can become ambiguous.

Recently, a class of numerical technique called the Immersed Interface Method (IIM) is developed to simulate the motion of a neutral fluid separated into two regions by a moving surface exerting forces. (See Xu and Wang [5] and references therein.) The IIM is a variant of the immersed boundary method, which was originally developed by Peskin[6] to simulate blood flows and has become one of the most powerful tool to solve fluid flows with material boundaries. The basic ideas of the IIM are (1)to utilize the generalized Taylor expansion[4] and (2)to derive the jump conditions required in (1). A schematic figure is shown in Fig.1. In a computational box, we have a material surface S. The dashed lines are some typical grid lines. Utilizing the generalized Taylor expansion, derivatives of a physical quantity $u(x)$ can be approximated in a manner similar to the standard finite difference method, provided that the jump conditions for $u(x)$ and its derivatives on the material surface[1] are already derived. Then the motion of the fluids is marched by standard techniques such as the Runge-Kutta-Scheme and the motion of the discontinuous surface can be tracked by a Lagrangean technique.

2 Jump conditions

While IIM is an excelent technique, at present its application is limited to the incompressible Navier-Stokes equation (for neutral fluids). One important step to make the technique applicable to fusion plasma simulations is to derive jump conditions for various physical quantities (including magnetic field and current density) in the case where the viscosity and the mass density are dicontinuous accross a surface. In this article, we present the jump conditions for velocity, pressure, and their derivatives in such a case. For simplicity, we restrict ourselves to the three-dimensional incompressible Navier-Stokes (NS)

author's e-mail: harahap.caesar@LHD.nifs.ac.jp

[1]Note that the words *material surface* mean a flow does not go across it. The surface is not necessariry a wall but can be, for example, a fluid surface with surface tension.

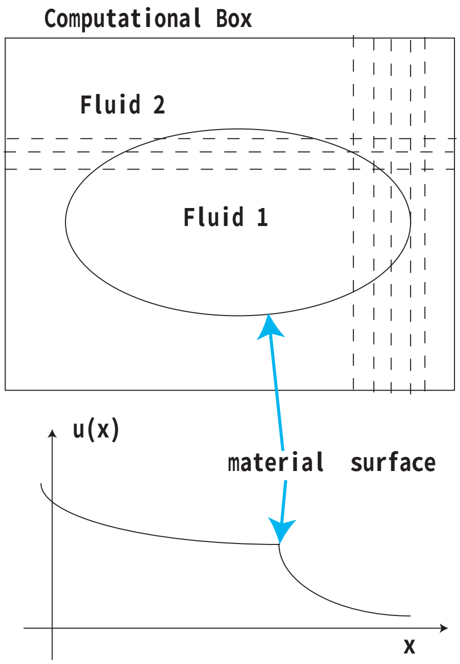

Fig. 1 A schematic view of two fluids separated by a material surface.

equations:

$$\rho\left(\frac{\partial u^i}{\partial t}+u^j\frac{\partial u^i}{\partial x_j}\right) = -\frac{\partial p}{\partial x^i}+\frac{\partial}{\partial x^i}\left\{\mu\left(\frac{\partial u^i}{\partial x^j}+\frac{\partial u^i}{\partial x^j}\right)\right\} + F^i, \quad (1)$$

$$\frac{\partial u^i}{\partial x^i} = 0, \quad (2)$$

$$F^i = \int_S f^i\left(\alpha^1,\alpha^2,t\right) \times \delta\left(\boldsymbol{x}-\boldsymbol{X}\left(\alpha^1,\alpha^2,t\right)\right)d\alpha^1 d\alpha 2, \quad (3)$$

$$\frac{\partial X^i\left(\alpha^1,\alpha^2,t\right)}{\partial t} = u^i\left(\boldsymbol{X}\left(\alpha^1,\alpha^2,t\right),t\right), \quad (4)$$

where (α^1,α^2) are Lagrangian parameters of points on the surface at a reference time, f^i is a Lagrangian force density (due to surface tension, for example), $\delta\left(\boldsymbol{x}-\boldsymbol{X}\left(\alpha^1,\alpha^2,t\right)\right)$ is a three dimensional delta function, and $\boldsymbol{X}\left(\alpha^1,\alpha^2,t\right)$ are Cartesian coordinates of the surface points. (See Fig.2). Taking the divergence of the momentum equation, a Poisson equation for pressure is obtained.

$$\frac{\partial^2 p}{\partial x^i\partial x^i} = \frac{\partial F^i}{\partial x^i}+2\frac{\partial^2}{\partial x^i\partial x^j}\left(\mu\frac{\partial u^i}{\partial x^j}\right) -\frac{\partial}{\partial x^i}\left\{\rho\left(\frac{\partial u^i}{\partial t}+u^j\frac{\partial u^i}{\partial x_j}\right)\right\}. \quad (5)$$

Differentiating $\boldsymbol{X}$ we obtain two tangent vectors along the coordinate lines of the surface.

$$\boldsymbol{\tau}_i = \frac{\partial \boldsymbol{X}}{\partial \alpha^i} \qquad (i=1,2). \quad (6)$$

The three components of the tangential vectors $\boldsymbol{\tau}_i$ in the Cartesian coordinates are denoted by $(\tau_i^1,\tau_i^2,\tau_i^3)$. The unit

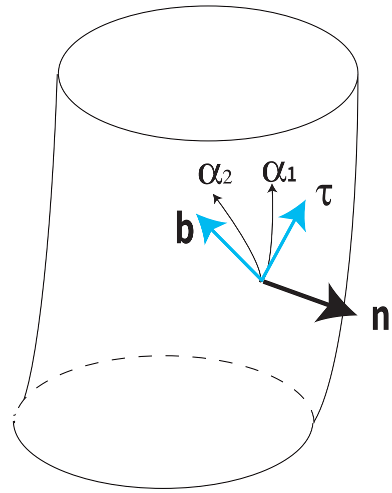

Fig. 2 Coordinate lines α^1, α^2 which cover the material surface and the tangential, normal and binormal unit vectors.

vector $\boldsymbol{n}$ normal to the surface (in Fig.1, pointing towards fluid 2) is given by

$$\boldsymbol{n} = \frac{1}{J}\left(\boldsymbol{\tau}_1\times\boldsymbol{\tau}_2\right), \quad (7)$$

$$J = \left|\boldsymbol{\tau}_1\times\boldsymbol{\tau}_2\right|. \quad (8)$$

We also introduce three mutually orthogonal unit vectors as $\boldsymbol{\tau}=\boldsymbol{\tau}_1/\left|\boldsymbol{\tau}_1\right|$, $\boldsymbol{n}$, $\boldsymbol{b}=\boldsymbol{n}\times\boldsymbol{\tau}$.

We assume that fluid 1 and fluid 2 have constant but different mass densities and viscosities. To show the jump conditions, we define some symbols here. A jump of a physical quantity accross the two phases is denoted by [.] where

$$[.]\left(\boldsymbol{X}(\alpha^1,\alpha^2,t),t\right) = (.)^+\left(\boldsymbol{X}(\alpha^1,\alpha^2,t),t\right) -(.)^-\left(\boldsymbol{X}(\alpha^1,\alpha^2,t),t\right). \quad (9)$$

The superscript $^+(^-)$ denotes the side towards which $\boldsymbol{n}$ points, while $^-$ denotes the other side.

The jump conditions are derived as follows. First, we formulate some basic jump coditions following the work by Xu and Wang[4]. No slip condition on the dicontinuous surface S leads to

$$\left[u^i\right] = 0. \quad (10)$$

and the acceleration jump condition

$$\left[\frac{\partial u^i}{\partial t}\right]+u^j\left[\frac{\partial u^i}{\partial x^j}\right]=0, \quad (11)$$

Extensions from the works by Xu and Wang[4, 5] appear in the jump conditions associated with the viscosity and the mass density. The jump condition for the viscous stress tensor in eq.(1) is given by

$$\left[\mu\left(\frac{\partial u^i}{\partial x^j}+\frac{\partial u^j}{\partial x^i}\right)\right]n^j = -\frac{f^i}{J}+\left(\frac{f^k n_k}{J}-2\left[\mu\right]\frac{\partial u^k}{\partial \tau}\tau_k -2\left[\mu\right]\frac{\partial u^k}{\partial b}b^k\right)n^i \quad (12)$$

The pressure jump condition is given by

$$[p] \quad = \quad \frac{f^i n_i}{J} - 2\,[\mu]\,\frac{\partial u^k}{\partial \tau}\tau_k - 2\,[\mu]\,\frac{\partial u^k}{\partial b}b^k, \qquad (13)$$

in which the jump of the viscosity is included. The jump condition for the pressure normal derivative

$$\begin{aligned}\left[\frac{\partial p}{\partial x_i}\right]n^i \quad &= \quad \frac{1}{J}\left[\frac{\partial \widetilde{f}^1}{\partial \alpha^1} + \frac{\partial \widetilde{f}^2}{\partial \alpha^2}\right] \\ &+2\,[\mu]\left(\frac{\partial^2 u^i}{\partial \tau^2}n^i + \frac{\partial^2 u^i}{\partial b^2}n^i + \frac{\partial u^i}{\partial \tau}\frac{\partial n_i}{\partial \tau} + \frac{\partial u^i}{\partial b}\frac{\partial n_i}{\partial b}\right) \\ &- \quad \left(\left[\rho\frac{\partial u^i}{\partial t}\right] + u^j\left[\rho\frac{\partial u^i}{\partial x_j}\right]\right)n^i. \qquad (14)\end{aligned}$$

contains both the viscosity jump and the mass density jump, making the formulation much more complicated than those in Refs.[4, 5].

After some manipulations, we have a set of coupled equations for the jump conditions as

$$\begin{pmatrix} \tau_1^1 & \tau_1^2 & \tau_1^3 & 0 & 0 & 0 & 0 & 0 & 0 \\ \tau_2^1 & \tau_2^2 & \tau_2^3 & 0 & 0 & 0 & 0 & 0 & 0 \\ 2n^1 & n^2 & n^3 & n^2 & 0 & 0 & n^3 & 0 & 0 \\ 0 & 0 & 0 & \tau_1^1 & \tau_2^1 & \tau_3^1 & 0 & 0 & 0 \\ 0 & 0 & 0 & \tau_1^2 & \tau_2^2 & \tau_3^2 & 0 & 0 & 0 \\ 0 & n^1 & 0 & n^1 & 2n^2 & n^3 & 0 & n^3 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & \tau_1^1 & \tau_2^1 & \tau_3^1 \\ 0 & 0 & 0 & 0 & 0 & 0 & \tau_1^2 & \tau_2^2 & \tau_3^2 \\ 0 & 0 & n^1 & 0 & 0 & n^2 & n^1 & n^2 & 2n^3 \end{pmatrix} \cdot \begin{pmatrix} \left[\mu\frac{\partial u^1}{\partial x^1}\right] \\ \left[\mu\frac{\partial u^1}{\partial x^2}\right] \\ \left[\mu\frac{\partial u^1}{\partial x^3}\right] \\ \left[\mu\frac{\partial u^2}{\partial x^1}\right] \\ \left[\mu\frac{\partial u^2}{\partial x^2}\right] \\ \left[\mu\frac{\partial u^2}{\partial x^3}\right] \\ \left[\mu\frac{\partial u^3}{\partial x^1}\right] \\ \left[\mu\frac{\partial u^3}{\partial x^2}\right] \\ \left[\mu\frac{\partial u^3}{\partial x^3}\right] \end{pmatrix}$$

$$= \begin{pmatrix} [\mu]\,\frac{\partial u^1}{\partial \alpha^1} \\ [\mu]\,\frac{\partial u^1}{\partial \alpha^2} \\ r_u^1 \\ [\mu]\,\frac{\partial u^2}{\partial \alpha^1} \\ [\mu]\,\frac{\partial u^2}{\partial \alpha^2} \\ r_u^2 \\ [\mu]\,\frac{\partial u^3}{\partial \alpha^1} \\ [\mu]\,\frac{\partial u^3}{\partial \alpha^2} \\ r_u^3 \end{pmatrix}, \qquad (15)$$

and

$$\begin{pmatrix} \tau_1^1 & \tau_2^1 & \tau_3^1 \\ \tau_1^2 & \tau_2^2 & \tau_3^2 \\ n^1 & n^2 & n^3 \end{pmatrix}\begin{pmatrix} \left[\frac{\partial p}{\partial x^1}\right] \\ \left[\frac{\partial p}{\partial x^2}\right] \\ \left[\frac{\partial p}{\partial x^3}\right] \end{pmatrix} \quad = \quad \begin{pmatrix} \frac{\partial r_p}{\partial \alpha^1} \\ \frac{\partial r_p}{\partial \alpha^2} \\ r_{pm} \end{pmatrix}, \qquad (16)$$

where

$$R_p \quad = \quad \frac{f^i n_i}{J} - 2\,[\mu]\,\frac{\partial u^k}{\partial \tau}\tau_k - 2\,[\mu]\,\frac{\partial u^k}{\partial b}b^k, \qquad (17)$$

$$r_u^i \quad = \quad -\frac{f^i}{J} + R_p n^i, \qquad (18)$$

$$\begin{aligned} r_{pn} \quad &= \quad \frac{1}{J}\left[\frac{\partial \widetilde{f}^1}{\partial \alpha^1} + \frac{\partial \widetilde{f}^2}{\partial \alpha^2}\right] \\ &+2\,[\mu]\left(\frac{\partial^2 u^i}{\partial \tau^2}n^i + \frac{\partial^2 u^i}{\partial b^2}n^i + \frac{\partial u^i}{\partial \tau}\frac{\partial n_i}{\partial \tau} + \frac{\partial u^i}{\partial b}\frac{\partial n_i}{\partial b}\right) \\ &- \quad \left(\left[\rho\frac{\partial u^i}{\partial t}\right] + u^j\left[\rho\frac{\partial u^i}{\partial x_j}\right]\right)n^i. \qquad (19)\end{aligned}$$

One way to utilize eqs.(15) and (16) is by giving the source terms in the right-hand-side by some appropriate interpolations or combinations of extrapolations from the two sides of each phases, forming a closed system of equations. Then the jump conditions can be obtained by inverting coefficient matrices in eqs.(15) and (16), enabling us to discretize the NS equations by the IIM method, and solve problems of multiphase flows with singular forces.

Jump conditions for $\left[\mu\partial^2 u^i/\partial x^j \partial x^k\right]$ are given as follows.

$$\begin{aligned}\begin{pmatrix} C_{11} & C_{12} & C_{13} \\ C_{21} & C_{22} & C_{23} \\ C_{31} & C_{32} & C_{33} \end{pmatrix}\begin{pmatrix} X_1 \\ X_2 \\ X_3 \end{pmatrix} \quad &= \quad [\mu]\begin{pmatrix} A_1 \\ A_2 \\ A_3 \end{pmatrix} \\ -\begin{pmatrix} B_{11} & B_{12} & B_{13} \\ B_{21} & B_{22} & B_{23} \\ B_{31} & B_{32} & B_{33} \end{pmatrix}&\begin{pmatrix} D_1 \\ D_2 \\ D_3 \end{pmatrix} + \begin{pmatrix} E_1 \\ E_2 \\ E_3 \end{pmatrix} \qquad (20)\end{aligned}$$

Since the expressions of the block matrices are lengthy, here we show only the first line of the expression:

$$X_1 \quad = \quad \begin{pmatrix} \left[\mu\frac{\partial^2 u^1}{\partial x^1 \partial x^1}\right] \\ \left[\mu\frac{\partial^2 u^1}{\partial x^1 \partial x^2}\right] \\ \left[\mu\frac{\partial^2 u^1}{\partial x^1 \partial x^3}\right] \\ \left[\mu\frac{\partial^2 u^1}{\partial x^2 \partial x^2}\right] \\ \left[\mu\frac{\partial^2 u^1}{\partial x^2 \partial x^3}\right] \\ \left[\mu\frac{\partial^2 u^1}{\partial x^3 \partial x^3}\right] \end{pmatrix}, \qquad (21)$$

$$C_{11} \quad = \quad \left(\begin{matrix} 2\,\left(\tau_1^1\right)^2 & 2\,\tau_1^1\tau_1^2 & 2\,\tau_1^1\tau_1^3 \\ 2\,\tau_1^1\tau_2^1 & \tau_1^1\tau_2^2 + \tau_1^2\tau_2^1 & \tau_1^1\tau_2^3 + \tau_1^3\tau_2^1 \\ 2\,\left(\tau_2^1\right)^2 & 2\,\tau_2^1\tau_2^2 & 2\,\tau_2^1\tau_2^3 \\ 3\,n^1\tau_1^1 & 2\,n^1\tau_1^2 + n^2\tau_1^1 & 2\,n^1\tau_1^3 + n^3\tau_1^1 \\ 3\,n^1\tau_2^1 & 2\,n^1\tau_2^2 + n^2\tau_2^1 & 2\,n^1\tau_2^3 + n^3\tau_2^1 \\ 1 & 0 & 0 \end{matrix}\right.$$

$$\left.\begin{matrix} 2\,\left(\tau_1^2\right)^2 & 2\,\tau_1^2\tau_1^3 & 2\,\left(\tau_1^3\right)^2 \\ 2\,\tau_1^2\tau_2^2 & \tau_1^2\tau_2^3 + \tau_1^3\tau 2^2 & 2\,\tau_1^3\tau_2^3 \\ 2\,\left(\tau_2^2\right)^2 & 2\,\tau_2^2\tau_2^3 & 2\,\left(\tau_2^3\right)^2 \\ 2\,n^2\tau_1^2 & n^2\tau_1^3 + n^3\tau_1^2 & 2\,n^3\tau_1^3 \\ 2\,n^2\tau_2^2 & n^2\tau_2^3 + n^3\tau_2^2 & 2\,n^3\tau_2^3 \\ 1 & 0 & 1 \end{matrix}\right) \qquad (22)$$

$$C_{12} \quad = \quad \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ n^2\tau_1^1 & n^2\tau_1^2 & n^2\tau_1^3 & 0 & 0 & 0 \\ n^2\tau_2^1 & n^2\tau_2^2 & n^2\tau_2^3 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}, \qquad (23)$$

$$C_{13} \quad = \quad \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ n^3\tau_1^1 & n^3\tau_1^2 & n^3\tau_1^3 & 0 & 0 & 0 \\ n^3\tau_2^1 & n^3\tau_2^2 & n^3\tau_2^3 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}, \qquad (24)$$

$$A_1 = \begin{pmatrix} \frac{\partial^2 u^1}{\partial\alpha^1\partial\alpha^1} \\ \frac{\partial^2 u^1}{\partial\alpha^1\partial\alpha^2} \\ \frac{\partial^2 u^1}{\partial\alpha^2\partial\alpha^2} \\ 0 \\ 0 \\ 0 \end{pmatrix}, \tag{25}$$

$$D_1 = \begin{pmatrix} \left[\mu\frac{\partial u^1}{\partial x^1}\right] \\ \left[\mu\frac{\partial u^1}{\partial x^2}\right] \\ \left[\mu\frac{\partial u^1}{\partial x^3}\right] \\ 0 \\ 0 \\ 0 \end{pmatrix}, \tag{26}$$

$$B_{11} = \begin{pmatrix} \frac{\partial\tau_1^1}{\partial\alpha^1} & \frac{\partial\tau_1^2}{\partial\alpha^1} & \frac{\partial\tau_1^3}{\partial\alpha^1} & 0 & 0 & 0 \\ \frac{\partial\tau_1^1}{\partial\alpha^2} & \frac{\partial\tau_1^2}{\partial\alpha^2} & \frac{\partial\tau_1^3}{\partial\alpha^2} & 0 & 0 & 0 \\ \frac{\partial\tau_2^1}{\partial\alpha^2} & \frac{\partial\tau_2^2}{\partial\alpha^2} & \frac{\partial\tau_2^3}{\partial\alpha^2} & 0 & 0 & 0 \\ 2\frac{\partial n^1}{\partial\alpha^1} & \frac{\partial n^2}{\partial\alpha^1} & \frac{\partial n^3}{\partial\alpha^1} & 0 & 0 & 0 \\ 2\frac{\partial n^1}{\partial\alpha^2} & \frac{\partial n^2}{\partial\alpha^2} & \frac{\partial n^3}{\partial\alpha^2} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}, \tag{27}$$

$$B_{12} = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ \frac{\partial n^2}{\partial\alpha^1} & 0 & 0 & 0 & 0 & 0 \\ \frac{\partial n^2}{\partial\alpha^2} & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}, \tag{28}$$

$$B_{13} = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ \frac{\partial n^3}{\partial\alpha^1} & 0 & 0 & 0 & 0 & 0 \\ \frac{\partial n^3}{\partial\alpha^2} & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}, \tag{29}$$

$$E_i = \begin{pmatrix} 0 \\ 0 \\ 0 \\ \frac{\partial r_u^i}{\partial\alpha^1} \\ \frac{\partial r_u^i}{\partial\alpha^2} \\ E_i^6 \end{pmatrix}, \tag{30}$$

$$E_i^6 = \left[\frac{\partial p}{\partial x^i}\right] + \left[\rho\frac{\partial u^i}{\partial t}\right] + u^1\left[\rho\frac{\partial u^i}{\partial x^1}\right] + u^2\left[\rho\frac{\partial u^i}{\partial x^2}\right] + u^3\left[\rho\frac{\partial u^i}{\partial x^3}\right]. \tag{31}$$

The other block matrices are also derived easily. The jump conditions for $\left[\frac{\partial^2 p}{\partial x^j \partial x^k}\right]$ are given in similar expressions.

3 Concluding Remarks

We derived the jump conditions in multiphase Navier-Stokes flows with singular forces, the phases having different mass densities and viscosities. Numerical computations utilizing these formulations will be shown in our future work. To make the IIM method applicable to MHD simulations or other fluid models of fusion plasmas, jump conditions for many variables and coupled combinations of the variables must be formulated.

[1] H. Miura et al., J. Plasma Physics **72**, 1095 (2006).
[2] H. Miura et al., Fusion Science and Technology **51**, 8 (2007).
[3] T.-H. Watanabe et al., Nucl. Fusion **47**, 1383 (2007).
[4] S. Xu and Z. Jane Wang, SIAM J. Sci. Comput. **27**, 1948 (2006).
[5] S. Xu and Z. Jane Wang, Comput. Methods Appl. Mech. (2007), doi:10.10.16/j.cma.2007.06.012.
[6] C.S. Peskin, J. Comput. Phys. **25**, 220 (1977).