Macromodeling for Microfluidic Channels

Chi-Wei Kuo, Chih-Ming Chien and Yao-Joe Yang
Department of Mechanical Engineering
*Institute of Applied Mechanics
National Taiwan University, Taipei, Taiwan, ROC

Abstract

In this paper, we present a methodology of generating microfluidic macromodels for 3-D non-linear microchannels using the *proper orthogonal decomposition* (POD) as well as the Galerkin method. The generated macromodels consists of basis functions and a low-order ODE model. In order to generate the basis functions, the ensembles of snapshots of fluidic field distributions are calculated by a FEM/FVM solver. Then the POD was employed to extract the basis functions from the ensembles. The Galerkin condition was adopted to create the low-order ODE model. Compared with the full-meshed simulations, the generated macromodels not only provide accurate results, but also give three order-of-magnitude speed-ups. Also, we demonstrate that it is possible to create the macromodel of a complicated channel by assembling the low-order ODE models and the basis functions of other simple channels.

Keywords: macromodel, microfluidic, POD, Galerkin method

1 Introduction

There are numerous applications in microfluidic systems, such as DNA sample detection, cell separation, μPCR, micromixer, and so on [1]. Microchannel is one of the most fundamental building blocks for microfluidic devices. The typical modeling tools for microchannels are the finite-element (FEM) or the finite-volume (FVM) solvers. These tools, although commercially available, usually require significant computational resources for the devices with complicated 3D geometries. Furthermore, if any design parameters are changed, the same computational resources are needed again for new computations. In this work, we develop a methodology that employs a model-order-reduction (MOR) technique [2-4] to generate accurate time-domain macromodels from 3-D full-meshed simulations. The microfluidic governing equation, the Navier-Stokes equation, is reduced to compact nonlinear low-order ordinary differential equations using the Galerkin condition. The velocity bases are generated from the snapshots of the CFDRC® simulations by using the proper orthogonal decomposition (POD) [5]. The flowchart of the procedure is shown in Figure 1. The macromodeling of the simple rectangular channel (SRC) and the L-shaped channel (LSC) will be studied thoroughly in this work. The accuracy as well as speed-ups of their corresponding macromodels will also be provided. We will demonstrate that the macromodel of a very complicated microchannel can be built by assembling the macromodels of a few basic building blocks. The assembling process includes two steps: (1) concatenation of basis functions and (2) combination of system matrices. The macromodels of microchannels possess modularized property that enables efficient system-level design for microfluidic systems.

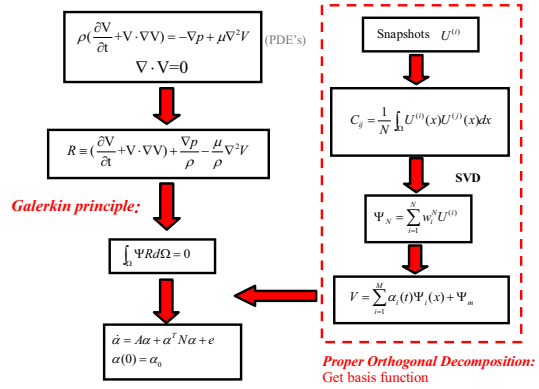

Figure 1: The procedure of the macromodeling

2 Macromodeling of Microfluidic Systems

The governing equation used in this work is the Navier-Stokes equation and the continuity equation :

$$\frac{\partial V}{\partial t}+V\cdot\nabla V-\frac{\mu}{\rho}\Delta V+\frac{1}{\rho}\nabla P=0 \quad (1)$$

$$\nabla V=0 \quad (2)$$

where ρ is the density of the fluid, V is the velocity vector field, g is the acceleration gravity, P is the pressure, and μ is the viscosity coefficient. Note that here we assume that the fluid is incompressible and the gravity effect is negligible.

For using the Galerkin method to reduce the fluidic governing equations into low-order models, the velocity/pressure distribution are approximated as the superposition of the velocity/pressure basis function Ψ_i and the time-depend coefficients $\alpha_i(t)$:

$$\hat{\Upsilon}=\sum_{i=1}^{n}\alpha_i(t)\Psi_i \quad (3)$$

where t is the time variable, n is the total number of the snapshots. Assume Equations (1) and (2) can be written as the following forms:

$$L(\Upsilon)=f \quad (4)$$

where L is either the differential operator for the Navier-Stokes equation or the continuity equation, and Υ is the exact solution of fields of the partial differential equations. In this case, Υ is the velocity distribution or the pressure distribution.

The Galerkin condition requires that

$$(B_i, L(\hat{\Upsilon})-f)=0 \quad (5)$$

Therefore, Equation (1) and (2) can be rearranged as a set of ordinary differential equations that is reduced from the original partial differential equations:

$$\begin{bmatrix}[M] & 0\\ 0 & 0\end{bmatrix}\begin{Bmatrix}\{\dot{\alpha}\}\\ \{\dot{\alpha}^P\}\end{Bmatrix}+\begin{bmatrix}[A] & 0\\ 0 & [Q]\end{bmatrix}\begin{Bmatrix}\{\alpha\}\\ \{\alpha^P\}\end{Bmatrix}+\begin{Bmatrix}\{N\alpha\alpha\}\\ 0\end{Bmatrix}+\begin{Bmatrix}\{e\}\\ 0\end{Bmatrix}=\begin{Bmatrix}0\\ 0\end{Bmatrix} \quad (6)$$

where M is the mass matrix, N is the convection matrix, A is the diffusion matrix, e is the effect of the outside force and Q is the continuity matrix. The details of these matrices can be found in Appendix I.

The time-dependent coefficient vector α in Equation (6) can be easily solved by time integration. The detailed field distributions (i.e., velocity and pressure) for each time step can be explicitly evaluated using Equation (3) once α is obtained. In Section IV, we will present a few microfluidic modeling cases using this macromodeling technique.

3 Proper Orthogonal Decomposition

In this section, we will describe the fundamentals of using the proper orthogonal decomposition (POD) technique to extract the basis functions for nonlinear micro fluid systems. This method is desirable for microfluidic modeling because it yields a small set of orthogonal basis functions that are capable of representing the physical field distributions of nonlinear systems when used with the Galerkin method. The brief description is as follows.

Considering a set of vector fields $U^{(i)}$ which is an ensemble of snapshots of a dynamical system's behavior. Assuming that a vector field Ψ that can approximate the vector fields $U^{(i)}$ as much as possible. Therefore, we may define a parameter λ as:

$$\lambda = \frac{1}{n}\sum_{i=1}^{n}\frac{\left|(U^{(i)},\Psi)\right|^2}{(\Psi,\Psi)} \tag{7}$$

where i is the index for the *i-th* snapshot, n is the total number of the snapshots. Note that here (f,g) is the inner product of two functions *f* and *g*.

Mathematically, Equation (7) is the average of the normalized inner product of $U^{(i)}$ and Ψ. Obviously, the larger the λ, the better the vector field Ψ approximates the set of the U fields. Assume that Ψ is a linear combination of the snapshots $U^{(i)}$:

$$\Psi = \sum_{i=1}^{n} w_i U^{(i)} \tag{8}$$

Substituting Equation (8) into (7), the max λ can be obtained by varying the weighting coefficients w_i. In order to simplify the derivation, we define an operator K as:

$$K\Psi = \frac{1}{n}\sum_{i=1}^{n}\int_{\Omega} U^{(i)}(x)U^{(i)}(x')\Psi(x')dx' \tag{9}$$

Therefore,

$$(K\Psi,\Psi) = \frac{1}{n}\sum_{i=1}^{n}\int_{\Omega}\int_{\Omega} U^{(i)}(x)\Psi(x)U^{(i)}(x')\Psi(x')dxdx' = \frac{1}{n}\sum_{i=1}^{n}\left|(U^{(i)},\Psi)\right|^2 \tag{10}$$

And thus Equation (7) can be written as:

$$\lambda = \frac{1}{n}\sum_{i=1}^{n}\frac{\left|(U^{(i)},\Psi)\right|^2}{(\Psi,\Psi)} = \frac{(K\Psi,\Psi)}{(\Psi,\Psi)} \tag{11}$$

We assume that Ψ^* is corresponding to the maximum value of λ. After rearranging Equation (11), we obtain the following eigenvalue problem:

$$K\Psi^* = \lambda\Psi^* \tag{12}$$

Using Equation (8), the eigenvalue problem can be written in terms of the weighting coefficients:

$$C_{ij}W_j = \lambda W_i \tag{13}$$

where $C_{ij} = \frac{1}{n}\int_{\Omega} U^{(i)}(x)U^{(j)}(x)dx$ and $W_i = \{w_1^i \quad w_2^i \quad \cdots \quad w_n^i\}^T$. In Equation (13), C_{ij} is the spatial correlation matrix in which i and j are corresponding to the *i-th* and *j-th* snapshots, respectively. W_i is the eigen-vector of the spatial correlation matrix that contains the weighting coefficients for the specific eigen-value λ. The eigen-values of the eigen-problem (Equation 13) are arranged as. $\lambda_1 \ge \lambda_2 \ge \lambda_3 \ge \cdots \ge \lambda_n$. The basis functions can be written as:

$$\Psi_1 = \sum_{i=1}^{n} w_i^1 U^{(i)}\ ,\ \Psi_2 = \sum_{i=1}^{n} w_i^2 U^{(i)}\ ,\ \cdots\ ,\ \Psi_n = \sum_{i=1}^{n} w_i^n U^{(i)} \tag{14}$$

Ψ_p is the sub-space of the POD, where $p = 1,2\ldots n$. Also, Ψ_p is in fact the linear combination of the snapshots with the weighting coefficients solved by the eigen-problem, and also known as the basis function of the system.

4 Case Studies

In this section, we will present the macromodeling results for two types of microchannels with different geometrical complexities. The CFDRC® is employed to generate the velocity and pressure snap-shots that will be used to obtain the basis functions by the POD.

4.1 Simple Rectangular Channel(SRC)

The first modeling case is the SRC, whose schematic view is shown in Figure 2. The typical dimensions of cross-section for microchannels are about 100 μm, and the dimensions for this case are also shown in Figure 2. The pressure boundary condition on the inlet surface is P_0, and the pressure is zero on the surface of outlet. The velocity on the channel walls is zero. Note that P_0 is the prescribed variable that can be a function of time and spatial coordinates.

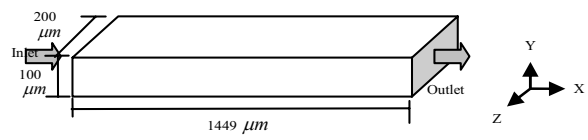

Figure 2: The model of the simple rectangular channel

4.1.1 Preliminary Macromodeling Results

We assume that the fixed input pressure $P_0 = 5\,N/m^2$. The fluidic properties are listed in Table 1. A transient CFDRC simulation is performed until steady state is reached. The snapshots of the field distributions are extracted from the simulations at specific time steps. For this case, the total number of snapshots (time steps) is 100. Figure 3 shows the normalized basis functions generated from the snapshots of CFDRC runs using the POD.

Density	$\rho = 999.82\,kg/m^3$
viscosity coefficient	$\mu = 0.001002\,N.s/m^2$
dynamic viscosity coefficient	$v = 1.004\times10^{-6}\ m^2/s$

Table 1: the parameter of the fluid

Using the basis functions as well as the macromodel that can be easily evaluated using Equation (6), the transient simulated results of the macromodels with different orders, as well as the transient full-meshed CFDRC results, are shown in Figure 4. The local errors between the macromodels and the full-meshed (CFDRC) model are less than 1% for the order greater than 3.

Obviously, increasing the order of the macromodel can effectively increase the accuracy of the models. The transient error is relatively larger when the fluid is still unsteady ($t < 0.002$), while the steady-state error is less than 1% even for the case with order 1.

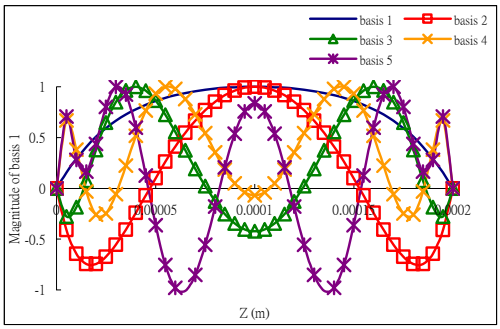

Figure 3: The basis function of the SRC

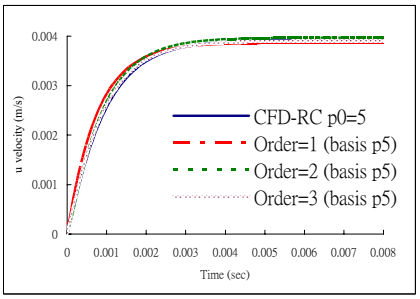

Figure 4: The average velocity of the macromodel for SRC

4.1.2 The Reusability of Basis Functions

Generating the snapshots of field distributions for extracting basis functions requires considerable computational resource. Therefore, for the same geometrical solid model, it will be very advantageous if the basis functions extracted from the model of a certain boundary condition can also be used for the case with different boundary conditions. Figure 5 shows the transient error between the macromodels and CFDRC runs using two sets of basis functions that were generated by two different CFDRC runs ($P_0 = 5N/m^2$ and $P_0 = 50N/m^2$). The negligible discrepancy in the figure indicates that the basis functions can be reusable for different inlet pressure (driving) conditions.

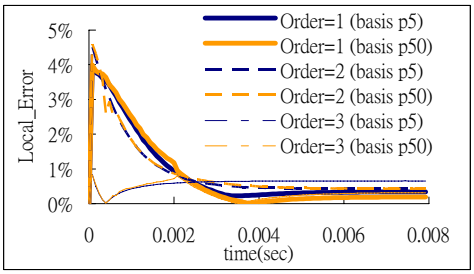

Figure 5: The error of the macromodels using the basis functions extracted from the CFDRC runs with different inlet pressure

Also, we will demonstrate that the basis functions extracted from the fixed inlet pressure can be used for the case with time-varying inlet pressure. For this case, the pressure on the inlet surface is represented as a time-varying function $P_0 = 50(1-\cos w_c t)$, where w_c is $833.3 rad/s$. The fluidic properties are the same as those of previous cases. Using the same basis functions generated from the fixed-inlet-pressure case, the simulated transient results of the macromodel are shown in Figure 6. Note that the macromodel of this case in exact the same as the fixed-inlet-pressure case except that the driving term uses time-varying P_0 (see Equation A1). The discrepancy between the results of the macromodels and the full-meshed model is less than 2%, and the steady error is less than 1%.

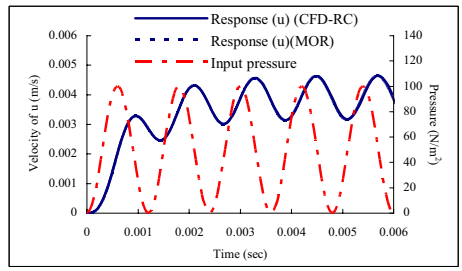

Figure 6: The velocity u of the macromodel as the input pressure changes sinusoidally with time

4.2 Macromodeling of L-Shaped Channels (LSC)

LSC is also one of the building blocks for microfluidic systems. The dimension of the LSC is shown in Figure 7. The pressure of the boundary condition on the inlet surface is P_0, and the pressure is zero on the outlet. The velocity on the channel walls is zero. The fluidic properties are the same as those of the simple-rectangular-channel case, and the input pressure is $P_0 = 5N/m^2$. Figure 8 shows the transient errors of the macromodels with different orders. Similar to the SRC, the error is relatively large when the fluid is still unsteady, while the error is negligible when the fluid reaches steady state. The macromodels with order greater than 2 give excellent results.

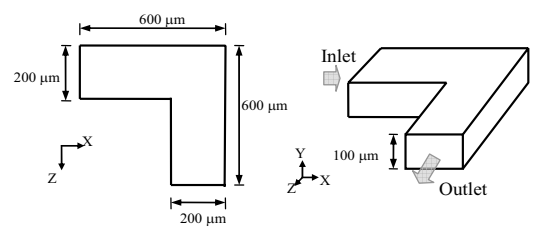

Figure 7: The model of LSC

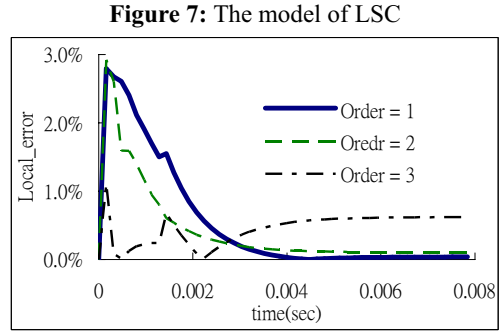

Figure 8: The transient error of the L-shaped channel

4.3 Computational Cost of Macromodels

The comparison between the computational times of the macromodels and the full-meshed CFDRC runs is listed in Table 2. This table also indicates the computational cost for generating macromodels (i.e., POD and Galerkin differentiation/integration calculations). It has to be emphasized that the cost listed in Table 2(a) and Table 2(b) are in fact the one time cost during the macromodel generating process because macromodels are reusable. Obviously, since the macromodels only employ a few basis functions, it is not surprised that the computational efficiency of macromodels are at least three -order-of-magnitude higher than the full-meshed models.

Channel	SRC	LSC
No. of Nodes	173061	607743
Simul time (=F)	69.3 min	35.4 hrs

Table 2(a): Results by CFD-RC (full-meshed model)

Channel	SRC			LSC		
Macromodel Order	1st	2nd	3rd	1st	2nd	3rd
POD compu. (sec)	7.5 (50 basis)			91.782 (10 basis)		
Compu. For $\partial\Psi/\partial x$	176.6			4277		
ODE matrix (sec)	1.1	1.4	2	5.8	7.4	10.6

Table 2(b): Macromodels Generation Cost

Channel	SRC			LSC		
Order	1st	2nd	3rd	1st	2nd	3rd
ODE comp. time	0.1	0.1	0.1	0.1	0.1	0.2
Simul. time (=R)	1.2	1.5	2.1	5.9	7.5	10.8
Speed-up (=F/R)	3365	2692	1923	21582	16978	11790

Table 2(c): Macromodel Simulation Cost

5 Assembling Macromodels

Each macromodel includes a basis function and a low-order ODE model. It is possible that the macromodel of a complicated microfluidic channel be easily created by combining the macromodel of other channels without performing expensive full-meshed FEM runs. The assembling procedure includes two steps: the first step is to concatenate the basis functions of the two modules. The second step is to combine the matrices of the macromodels to form a new macromodel. The two steps are described as follows:

(a) Concatenating Basis functions

Assuming that Channel A and Channel B will be assembled. The inlet of the new system is on Channel A. For each mode of the POD-extracted basis function, the normalized velocity profile on the outlet surface of Channel A and the normalized velocity profile on the inlet surface of Channel B are almost the same, since the extracted basis functions approximate fully-developed flow very well. Therefore, the basis functions of the two channels can be concatenated after certain normalization process.

(b) Constructing New Matrices of Low-order ODE Model

The Galerkin condition is in fact the inner product of the original governing equation and a corresponding basis function. Therefore, for the combined macromodel with certain order, the system matrices are actually the summation of the matrices of the original macromodels.

A complicated microchannel, as shown in Figure 9, can be easily built by the macromodels of fundamental building blocks. The serpentine structure consists of 8 LSC structures. Therefore, the system matrices of the low-order ODE for a specific order are:

$$M_{8L} = 8\times M_{LSC} \quad A_{8L} = 8\times A_{LSC} \quad N_{8L} = 8\times N_{LSC} \tag{15}$$

Figure 10 shows the transient local error between the assembled macromodel and the full-mesh calculation when the inlet pressure P_0 is $50\,N/m^2$. The errors are 1.9%, 1.7%, and 0.5% for the models with order 1, 2 and 3, respectively. These results indicate that it is possible to efficiently build the macromodel of a complicated channel from the macromodels of a few simple building blocks, without performing expensive full-meshed computations.

6 Conclusion

This paper presents a macromodeling technique using the proper orthogonal decomposition as well as the Galerkin method for 3-D non-linear microfluidic problems. The generated macromodel are a low-order ODE model incorporated with a few basis functions. Using the POD, the basis functions were extracted from the ensembles of snapshots of fluidic field distributions that were calculated by full-meshed FEM/FVM solvers. The low-order ODE model was created by the Galerkin condition. The simulated results not only indicated that the microfluidic behavior can be accurately reproduced using the macromodels, but also provided three-orders-of-magnitude speed-ups when compared with full-meshed computations. Also, the macromodel of a complicated channel can be built by assembling the low-order ODE models and the basis functions of various types of simple channels.

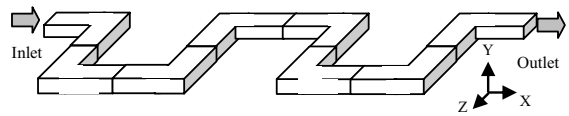

Figure 9: The schematic of a serpentine channel

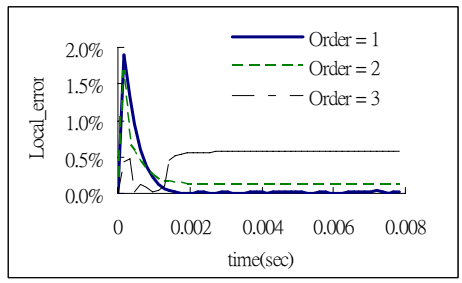

Figure 10: the transient error of the 8 L-shaped channel combination

Acknowledgement

This project is supported by the National Science Council, Taiwan (Contract No: 93-2213-E-002-038)

Reference

[1] L. M. Fu, C. H. Lin and G. L. Chang, "Multiple volume injection technique for high-resolution DNA sample detection utilizing planar microfluidic chip,", IEEE EMBS, Sep. 1-5, 2004, p.5017-5020.

[2] Y.-J. Yang and P.-C. Yen, "An Efficient Macromodeling Methodology for Lateral Air Damping Effects", IEEE/ASME JMEMS, Vol. 14, No. 4, pp. 812-828 (2005)

[3] H. M. Park and M. W. Lee, "Control of Navier-Stokes Equations by Means of Mode Reduction, "International Journal for Numerical Method in Fluids, Vol. 33, 2000, p535-557.

[4] S. S. Ravindran, "Proper Orthogonal Decomposition in Optimal Control of Fluids, "International Journal for Numerical Method in Fluids, Vol. 34, 2000, p425-448.

[5] Y.-J. Yang, S.-Y. Cheng and K.-Y. Shen, "Macromodeling of Coupled-Domain MEMS Devices with Electrostatic and Electrothermal Effects", JMM, Vol. 14, pp. 1191-1196 (2004)

Appendix A

$$M_{ij} = \int_\Omega \Psi_i^u\Psi_j^u + \Psi_i^v\Psi_j^v + \Psi_i^w\Psi_j^w \qquad N_{ijk} = N^{uuu} + N^{vu} + N^{wu} + N^{uv} + N^{vv} + N^{wv} + N^{uw} + N^{vw} + N^{ww}$$

$$N^{uuu} = \int_\Omega (\Psi_i^u\Psi_j^u\frac{\partial\Psi_k^u}{\partial x})d\Omega \quad N^{vu} = \int_\Omega (\Psi_i^v\Psi_j^u\frac{\partial\Psi_k^u}{\partial y})d\Omega \quad N^{wu} = \int_\Omega (\Psi_i^w\Psi_j^u\frac{\partial\Psi_k^u}{\partial z})d\Omega$$

$$N^{uv} = \int_\Omega (\Psi_i^u\Psi_j^v\frac{\partial\Psi_k^v}{\partial x})d\Omega \quad N^{vv} = \int_\Omega (\Psi_i^v\Psi_j^v\frac{\partial\Psi_k^v}{\partial y})d\Omega \quad N^{wv} = \int_\Omega (\Psi_i^w\Psi_j^v\frac{\partial\Psi_k^v}{\partial z})d\Omega$$

$$N^{uw} = \int_\Omega (\Psi_i^u\Psi_j^w\frac{\partial\Psi_k^w}{\partial x})d\Omega \quad N^{vw} = \int_\Omega (\Psi_i^v\Psi_j^w\frac{\partial\Psi_k^w}{\partial y})d\Omega \quad N^{ww} = \int_\Omega (\Psi_i^w\Psi_j^w\frac{\partial\Psi_k^w}{\partial z})d\Omega$$

$$A = A^u + A^v + A^w$$

$$A^u = \int_\Omega \frac{\mu}{\rho}(\frac{\partial\Psi_j^u}{\partial x}\frac{\partial\Psi_i^u}{\partial x} + \frac{\partial\Psi_j^u}{\partial y}\frac{\partial\Psi_i^u}{\partial y} + \frac{\partial\Psi_j^u}{\partial z}\frac{\partial\Psi_i^u}{\partial z})d\Omega + \oint_{\_inlet}\Psi_j^u(\frac{\mu}{\rho}\frac{\partial\Psi_i^u}{\partial x})dS - \oint_{\_outlet}\Psi_j^u(\frac{\mu}{\rho}\frac{\partial\Psi_i^u}{\partial x})dS$$

$$A^v = \int_\Omega \frac{\mu}{\rho}(\frac{\partial\Psi_j^v}{\partial x}\frac{\partial\Psi_i^v}{\partial x} + \frac{\partial\Psi_j^v}{\partial y}\frac{\partial\Psi_i^v}{\partial y} + \frac{\partial\Psi_j^v}{\partial z}\frac{\partial\Psi_i^v}{\partial z})d\Omega + \oint_{\_inlet}\Psi_j^v(\frac{\mu}{\rho}\frac{\partial\Psi_i^v}{\partial x})dS - \oint_{\_outlet}\Psi_j^v(\frac{\mu}{\rho}\frac{\partial\Psi_i^v}{\partial x})dS$$

$$A^w = \int_\Omega \frac{\mu}{\rho}(\frac{\partial\Psi_j^w}{\partial x}\frac{\partial\Psi_i^w}{\partial x} + \frac{\partial\Psi_j^w}{\partial y}\frac{\partial\Psi_i^w}{\partial y} + \frac{\partial\Psi_j^w}{\partial z}\frac{\partial\Psi_i^w}{\partial z})d\Omega + \oint_{\_inlet}\Psi_j^w(\frac{\mu}{\rho}\frac{\partial\Psi_i^w}{\partial x})dS - \oint_{\_outlet}\Psi_j^w(\frac{\mu}{\rho}\frac{\partial\Psi_i^w}{\partial x})dS$$

$$e = -\oint_{\_inlet}\Psi_j^u(\frac{1}{\rho}P_0)dS \qquad i,j,k = 1\cdots m \tag{A1}$$

where m is the total number of the basis functions for macromodels.