INTERNATIONAL JOURNAL FOR NUMERICAL METHODS IN FLUIDS
Int. J. Numer. Meth. Fluids 2008; **58**:119–159
Published online 7 January 2008 in Wiley InterScience (www.interscience.wiley.com). DOI: 10.1002/fld.1677

Shape optimization using adjoint variable method for reducing drag in Stokes flow

Kazunori Shinohara[1,*,†], Hiroshi Okuda[2], Satoshi Ito[3], Norihiro Nakajima[4] and Masato Ida[4]

[1] *Intelligent Modeling Laboratory* (*IML*), *University of Tokyo*, *2-11-16*, *Yayoi*, *Bunkyo-ku*, *Tokyo 113-8656*, *Japan*
[2] *Research into Artifacts*, *Center for Engineering* (*RACE*), *University of Tokyo*, *5-1-5 Kashiwanoha*, *Kashiwa Chiba 277-8568*, *Japan*
[3] *Institute of Industrial Science* (*IIS*), *University of Tokyo*, *Komaba Research Campus*, *4-6-1 Komaba Meguro-ku*, *Tokyo 153-8505*, *Japan*
[4] *Center for Computational Science and e-Systems* (*CCSE*), *Japan Atomic Energy Agency*, *6-9-3 Higashi-Ueno*, *Taitoku*, *Tokyo 110-0015*, *Japan*

SUMMARY

To minimize the drag of a 3D object placed under Stokes flow, an adjoint variable method based on the variational principle was formulated and applied to the finite element method. The optimality condition of the present method consists of the state equations, the adjoint equations, and the sensitivity equations. As related techniques, a smoothing method for the surface mesh, a method to guarantee the constant volume condition, and a mesh relocation technique using the biharmonic equation were developed. These techniques are crucial in realizing shape optimization. To overcome such difficulties as heavy computational burden and large memory requirements, the present system was implemented with the data compression technology supplied with the parallel software library HEC-MW. Also, by utilizing HEC-MW, the program was efficiently parallelized, and the number of program lines was dramatically reduced. By the proposed shape optimization method, the drag on a cylinder was reduced by about 25% under a Stokes flow condition. Copyright © 2008 John Wiley & Sons, Ltd.

Received 27 March 2007; Revised 13 October 2007; Accepted 20 October 2007

KEY WORDS: shape optimization; CFD; FEM; HEC-MW; adjoint variable method; Stokes flow; smoothing; mesh deformation

[*] Correspondence to: Kazunori Shinohara, Intelligent Modeling Laboratory (IML), University of Tokyo, 2-11-16, Yayoi, Bunkyo-ku, Tokyo 113-8656, Japan.

[†] E-mail: sinohara@nihonbashi.race.u-tokyo.ac.jp

Contract/grant sponsor: Japan Atomic Energy Agency
Contract/grant sponsor: Intelligent Modeling Laboratory
Contract/grant sponsor: HEC-MW group

1. INTRODUCTION

The goal of shape optimization is to transform the external surface of an object to obtain increased performance, high reliability, and low cost. Shape optimization in the present study means deriving the optimal shape to minimize (or maximize) a cost function under some constraints. Historically, the shape optimization technique has been employed in a wide range of fields. In the industrial products affecting fluids, the optimal shape for an airplane wing has been obtained by using sensitivity analysis [1]. Shape optimization has also been applied to the hull shape [2], the turbine wing shape [3], the horn shape, etc. [4]. These studies involved shape optimizations for large objects under unsteady flow. On the other hand, Stokes flow (i.e. steady, low Reynolds number flow) analyses such as those performed in microfluidics have been needed since the 1990s. These analyses are mainly used in the development of medical devices such as DNA chips [5], blood pumps [6, 7], dielectrophoretic devices [8], and grooved structures [9]. In microfluidics, fluids inside devices cannot be easily transported through the structure (e.g. channels) because of the traction generated by the wall. Also, fluids will not easily mix since turbulences are difficult to generate. Thus, the flow inside microdevices remains to be a Stokes flow and the Reynolds number is extremely low. In recent years, the building of optimal structures with respect to both fluid transportation and fluid mixture was investigated [10, 11]. However, such a kind of micro and nano system technology has not been fully developed yet. As experimental microfluidics and nanofluidics are very difficult to realize, theoretical and computational methods play a relatively dominant role at the present time. Stokes flow analyses have therefore become increasingly important.

In studies of the shape optimization under Stokes flow, shape optimization algorithms to minimize the drag on an object under a constant volume condition have been studied [12]. Sano and Sakai constructed a 2D optimal shape that minimized a cost function under a constant area by using the Stokes equation [13]. Bessho and Himeno presented an inverse procedure for minimization of the drag [14]. Taseli *et al.* employed a geometrical shape function to tackle the problem by using the simplex method [15]. Ganesh used Pironneau's algorithm to reduce the drag on an object and obtained an optimal shape [16]. Huan and Modi investigated 2D minimum-drag bodies for a range of Reynolds numbers varying from 20 to 100 000 [17]. Wagner *et al.* developed a numerical shape optimization tool for asymmetric minimum-drag bodies using evolution strategies [18]. Katamine and Azegami proposed the traction method and obtained the same optimal shape as that of Pironneau [19]. By using variational approaches, Richardson considered the problem of designing the section of a cylinder to minimize the drag on it [20]. Kim and Kim carried out shape optimization for low Reynolds numbers ($Re \ll 40$) by using the Navier–Stokes equations [21]. Datta and Srivastava constructed the maximum drag profile of an axis-symmetric body under Stokes flow [22]. Lund *et al.* presented a gradient-based shape optimization method for strongly coupled stationary fluid–structure interaction problems [23]. Using an adjoint variable method, Yagi and Kawahara solved steady and unsteady flow problems after a careful comparison of the optimal shape that they obtained with Pironueau's result [24]. Matsumoto used the bubble function in his shape optimization algorithm and solved unsteady flow problems [25]. Guest presented the topology optimization for objects in Stokes flows [26].

In the above-mentioned studies, various optimization methods, both probabilistic and deterministic, were used. Among the former, genetic algorithms (GAs) are mostly used, while,

DOI: 10.1002/fld

in the latter, the adjoint variable method seems to be the method of choice because it has the advantage of being able to obtain the sensitivities for all design variables from a single flow calculation. Regarding numerical methods, in the context of shape optimization of complex structures such as microdevices, discretization based on the finite element method (FEM) is recently more often employed than the one based on the finite difference method. Therefore, in the present study, we are based on the adjoint variable method and FEM.

In the shape optimization problem involving complex structures, complicated flow and large deformations of objects, combinations of both adjoint variable method and peripheral techniques are demanded. We identify five major issues in realizing a high-quality shape optimization based on the adjoint variable method.

The first issue is smoothing. As the object is deformed according to the sensitivity, the object surface that is initially smooth gradually becomes irregular [27]. This irregular surface partly causes negative volume elements in the finite element and consequently the fluid computation cannot be continued. When a low-resolution mesh is used the surface stays regular, as the number of control points (nodal points) is rather small. However, the accuracy of the optimal shape is rather low. For this reason, high-resolution meshes have to be employed. Such high-resolution meshes contain many control points that cause the shape to become irregular in the deformation process. Thus, a smoothing method for the surface mesh is necessary.

The second issue is large deformation of objects. When the object shape is largely deformed from the initial shape, some finite elements could have a negative volume in the shape deformation process. In consequence, robust mesh deformation becomes a necessity.

The third issue is the constraint of constant volume. According to sensitivities, the volume of objects finally becomes negative, leading to an unrealistic deformation. Therefore, techniques of applying the constant volume constraint are demanded on the real design.

The forth issue is parallel computing. High-resolution meshes should be necessary to obtain more accurate results and construct more detailed optimal shapes. Parallel techniques are naturally needed to distribute the large calculation load resulting from the use of a high-resolution mesh.

The fifth issue lies just in the adjoint variable method. To appropriately implement the above-mentioned techniques in the sensitivity analysis based on the adjoint variable method, both the calculation procedure and the theoretical derivation (to obtain the state, the adjoint, and the sensitivity equations) of this method should be specified. Fusion techniques to match these techniques with the adjoint variable method are demanded.

In this study, versatile combined techniques consisting of sensitivity analysis based on the FEM, smoothing, constant volume condition, robust mesh modification, and parallel computing using the HEC-MW [28] are proposed. First, the Laplace smoothing method is applied to the surface shape. Second, the biharmonic equation consisting of the fourth derivative is applied to deform the mesh. Following, an iterative control approach with respect to the constant volume is implemented. Next, the shape optimization algorithm is parallelized by utilizing parallel library HEC-MW. Last, the adjoint variable method containing the discretized approach based on FEM is presented. By using these combined techniques, 3D cylinder optimization, which is not conventionally performed, is tried. The effectiveness of the shape optimization algorithm is shown by comparing the present results with Pironneau's results [12].

2. ADJOINT VARIABLE METHOD

2.1. Definition

In Figure 1, the solid line and the dotted line schematically show the initial shape and the optimized shape in the fluid, respectively. We denote by Ω the computational domain. We denoted by Γ, γ, and Ψ the boundaries of the computational domain. We defined Γ as follows:

$$\Gamma = \Gamma_{\mathrm{E}} + \Gamma_{\mathrm{W}} + \Gamma_{\mathrm{S}} + \Gamma_{\mathrm{N}} + \Gamma_{\mathrm{U}} + \Gamma_{\mathrm{D}} + \gamma = \psi + \gamma \tag{1}$$

where subscripts E, W, S, N, U, and D indicate the boundary parts. We denote time and the 3D spatial coordinate vector as follows:

$$t \in \mathbf{R}^1 \tag{2}$$

$$\mathbf{X} = (x_1, x_2, x_3)^{\mathrm{T}} = (x, y, z)^{\mathrm{T}} \in \mathbf{R}^3 \quad \text{in } \Omega \tag{3}$$

The coordinates of the object are defined as follows:

$$\mathbf{S} = (s_1, s_2, s_3)^{\mathrm{T}} \in \mathbf{R}^3 \quad \text{on } \gamma \subset \Omega \tag{4}$$

A unit normal vector on the surface shape γ is defined as follows:

$$\mathbf{n}(\mathbf{S}) = (n_1(\mathbf{S}), n_2(\mathbf{S}), n_3(\mathbf{S}))^{\mathrm{T}} \in \mathbf{R}^3 \quad \text{on } \gamma \tag{5}$$

A unit normal vector on the boundary is also defined as follows:

$$\mathbf{n}(\mathbf{X}) = (n_1(\mathbf{X}), n_2(\mathbf{X}), n_3(\mathbf{X}))^{\mathrm{T}} \in \mathbf{R}^3 \quad \text{on } \Gamma_{\mathrm{N}}, \Gamma_{\mathrm{S}}, \Gamma_{\mathrm{W}}, \Gamma_{\mathrm{E}}, \Gamma_{\mathrm{U}}, \Gamma_{\mathrm{D}} \tag{6}$$

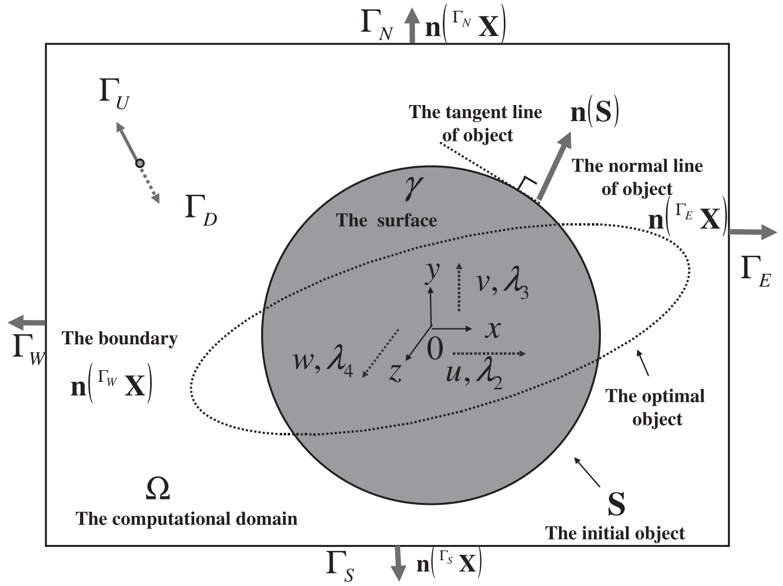


Figure 1. Variables and notations.

DOI: 10.1002/fld

The unity vector is used in the cost function. We define the velocity vector as follows:

$$\mathbf{u}(t,\mathbf{X})=(u_1(t,\mathbf{X}),u_2(t,\mathbf{X}),u_3(t,\mathbf{X}))^{\mathrm{T}}=(u(t,\mathbf{X}),v(t,\mathbf{X}),w(t,\mathbf{X}))^{\mathrm{T}}\in\mathbf{R}^3 \quad \text{in } \Omega \tag{7}$$

and the state variable vector is

$$\begin{aligned}\mathbf{W}(t,\mathbf{X})&=(W_1(t,\mathbf{X}),W_2(t,\mathbf{X}),W_3(t,\mathbf{X}),W_4(t,\mathbf{X}))^{\mathrm{T}}\\&=(p(t,\mathbf{X}),u(t,\mathbf{X}),v(t,\mathbf{X}),w(t,\mathbf{X}))^{\mathrm{T}}\in\mathbf{R}^4 \quad \text{in } \Omega\end{aligned} \tag{8}$$

where p denotes the pressure. The adjoint variable vector depending on time and spatial coordinates is defined as follows:

$$\boldsymbol{\lambda}(t,\mathbf{X})=(\lambda_1(t,\mathbf{X}),\lambda_2(t,\mathbf{X}),\lambda_3(t,\mathbf{X}),\lambda_4(t,\mathbf{X}))^{\mathrm{T}}\in\mathbf{R}^4 \quad \text{in } \Omega \tag{9}$$

where λ_1 represents the adjoint pressure and λ_2–λ_4 represents the adjoint velocity vectors. The superscript (n) shows the nth time step. The subscript $[i]$ shows the node number. The subscript $\langle m\rangle$ shows the element number. The subscript (k) shows the kth shape step. The shape step represents the number of shape modifications from the initial step to the optimal one. For example, the kth velocity of both the node $[i]$ and the element $\langle m\rangle$ in the time step (n) is defined as follows:

$$\begin{aligned}u^{(n)}_{(k)}&=(u^{(n)}_{1,(k),[i]}\ u^{(n)}_{2,(k),[i]}\ u^{(n)}_{3,(k),[i]})^{\mathrm{T}}\\&=(u^{(n)}_{(k),[i]}\ v^{(n)}_{(k),[i]}\ w^{(n)}_{(k),[i]})^{\mathrm{T}},\quad n,k,i=0,1,\ldots\in\mathbf{R}^3 \quad \text{in } \Omega\end{aligned} \tag{10}$$

$$\begin{aligned}u^{(n)}_{(k)}&=(u^{(n)}_{1,(k),\langle m\rangle}\ u^{(n)}_{2,(k),\langle m\rangle}\ u^{(n)}_{3,(k),\langle m\rangle})^{\mathrm{T}}\\&=(u^{(n)}_{(k),\langle m\rangle}\ v^{(n)}_{(k),\langle m\rangle}\ w^{(n)}_{(k),\langle m\rangle})^{\mathrm{T}},\quad n,k,m=0,1,\ldots\in\mathbf{R}^3 \quad \text{in } \Omega\end{aligned} \tag{11}$$

In the case that there is the same subscript i in the same term, the subscript i is the summation convention as follows:

$$a_ib_i=a_1b_1+a_2b_2+a_3b_3,\quad i=1,2,3 \tag{12}$$

2.2. Problem

To minimize the cost function under constraints, we formulated the Lagrange function by introducing the adjoint variables. The adjoint variable method is based on the variational method. By introducing Lagrange multipliers called adjoint variables, the constrained optimization of the cost function is transformed into the unconstrained optimization of the Lagrange function. A circular cylinder is placed in the computational domain Ω, as shown in Figure 2. Γ is the N–S–E–W boundary at north, south, east, and west. γ represents the surface of the object under optimization. A fluid flows in on the boundary Γ_w and flows out on the boundary Γ_{E}. The origin of coordinates is at the center of the cylinder.

In this paper, as the cost function, the traction force on the surface γ is defined as

$$J(\mathbf{S})=-\int_{t_{\mathrm{s}}}^{t_{\mathrm{e}}}\int_{\gamma(\mathbf{S})}\left\{\left(-\frac{p}{\rho}+2\nu\frac{\partial u}{\partial x}\right)n_1+\nu\left(\frac{\partial v}{\partial x}+\frac{\partial u}{\partial y}\right)n_2+\nu\left(\frac{\partial w}{\partial x}+\frac{\partial u}{\partial z}\right)n_3\right\}\mathrm{d}\gamma\,\mathrm{d}t\in\mathbf{R}^1 \tag{13}$$

The surface domain $\gamma(\mathbf{S})$ depends on $\mathbf{S}$. t_{s} and t_{e} show the start of the test time and the end of the test time in the optimization. We formulated the Lagrange function by introducing the adjoint

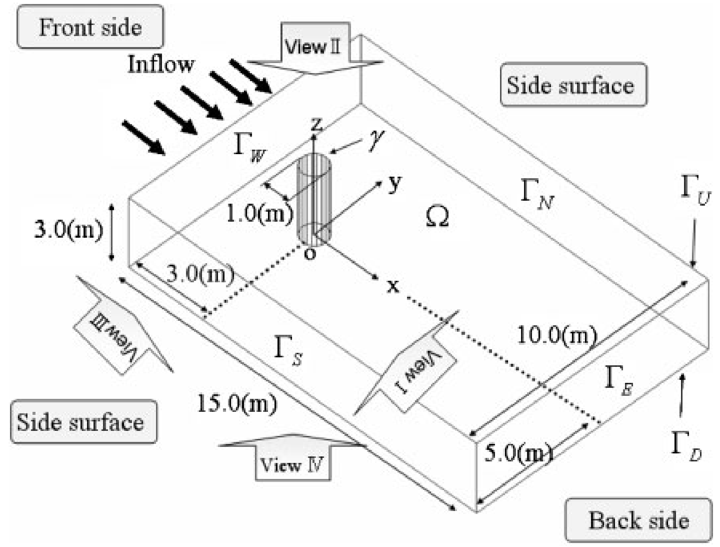


Figure 2. Computational domain and boundary conditions.

variable as follows:

$$
\begin{aligned}
L(\mathbf{S}) = J(\mathbf{S}) &+ \int_{t_\mathrm{s}}^{t_\mathrm{e}} \int_{\Omega(\mathbf{S})} \lambda_1 \frac{1}{\rho} \left(\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} + \frac{\partial w}{\partial z} \right) \mathrm{d}\Omega\, \mathrm{d}t \\
&+ \int_{t_\mathrm{s}}^{t_\mathrm{e}} \int_{\Omega(\mathbf{S})} \lambda_2 \left\{ -\frac{\partial u}{\partial t} - \frac{1}{\rho}\frac{\partial p}{\partial x} + 2v \frac{\partial}{\partial x}\frac{\partial u}{\partial x} + v \frac{\partial}{\partial y}\left(\frac{\partial v}{\partial x} + \frac{\partial u}{\partial y} \right) + v \frac{\partial}{\partial z}\left(\frac{\partial u}{\partial z} + \frac{\partial w}{\partial x} \right) \right\} \mathrm{d}\Omega\, \mathrm{d}t \\
&+ \int_{t_\mathrm{s}}^{t_\mathrm{e}} \int_{\Omega(\mathbf{S})} \lambda_3 \left\{ -\frac{\partial v}{\partial t} - \frac{1}{\rho}\frac{\partial p}{\partial y} + v \frac{\partial}{\partial x}\left(\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x} \right) + 2v \frac{\partial}{\partial y}\frac{\partial v}{\partial y} + v \frac{\partial}{\partial z}\left(\frac{\partial v}{\partial z} + \frac{\partial w}{\partial y} \right) \right\} \mathrm{d}\Omega\, \mathrm{d}t \\
&+ \int_{t_\mathrm{s}}^{t_\mathrm{e}} \int_{\Omega(\mathbf{S})} \lambda_4 \left\{ -\frac{\partial w}{\partial t} - \frac{1}{\rho}\frac{\partial p}{\partial z} + v \frac{\partial}{\partial x}\left(\frac{\partial u}{\partial z} + \frac{\partial w}{\partial x} \right) + v \frac{\partial}{\partial y}\left(\frac{\partial v}{\partial z} + \frac{\partial w}{\partial y} \right) + 2v \frac{\partial}{\partial z}\frac{\partial w}{\partial z} \right\} \mathrm{d}\Omega\, \mathrm{d}t \in \mathbf{R}^1
\end{aligned}
\tag{14}
$$

where λ_1 shows the adjoint pressure corresponding to the pressure p and λ_2–λ_4 show the adjoint velocity corresponding to the flow vector (u, v, w). The above equation can be transformed as follows (see Appendix A for details regarding this transformation):

$$
\begin{aligned}
L(\mathbf{S}) = J(\mathbf{S}) &+ \int_{t_\mathrm{s}}^{t_\mathrm{e}} \int_{\Gamma(\mathbf{S})} \lambda_{i+1} \left\{ -\frac{p}{\rho} n_i + v \left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} \right) n_j \right\} \mathrm{d}\Gamma\, \mathrm{d}t \\
&- \int_{t_\mathrm{s}}^{t_\mathrm{e}} \int_{\Gamma(\mathbf{S})} u_i \left\{ -\frac{1}{\rho} \lambda_1 n_i + v \left(\frac{\partial \lambda_{i+1}}{\partial x_j} + \frac{\partial \lambda_{j+1}}{\partial x_i} \right) n_j \right\} \mathrm{d}\Gamma\, \mathrm{d}t
\end{aligned}
$$

$$
\begin{aligned}
&+\int_{t_\mathrm{s}}^{t_\mathrm{e}}\int_{\Omega(\mathbf{S})} p\frac{1}{\rho}\left(\frac{\partial\lambda_2}{\partial x}+\frac{\partial\lambda_3}{\partial y}+\frac{\partial\lambda_4}{\partial z}\right)\mathrm{d}\Omega\,\mathrm{d}t\\
&+\int_{t_\mathrm{s}}^{t_\mathrm{e}}\int_{\Omega(\mathbf{S})} u_i\left\{\frac{\partial\lambda_{i+1}}{\partial t}-\frac{1}{\rho}\frac{\partial\lambda_1}{\partial x_i}+v\frac{\partial}{\partial x_j}\left(\frac{\partial\lambda_{j+1}}{\partial x_i}+\frac{\partial\lambda_{i+1}}{\partial x_j}\right)\right\}\mathrm{d}\Omega\,\mathrm{d}t\\
&-\int_{\Omega(\mathbf{S})}[\lambda_{i+1}u_i]_{t_\mathrm{s}}^{t_\mathrm{e}}\,\mathrm{d}\Omega,\quad i,j=1,2,3\in\mathbf{R}^1
\end{aligned}
\tag{15}
$$

In the above equation, the summation convention is used. The adjoint variable method consists of the state equation, the adjoint equation, and the sensitivity equation.

2.3. State equations

The state variable is calculated by solving the state equation, the stationary condition that is obtained by taking the derivative of the Lagrange function with respect to the adjoint variable $\boldsymbol{\lambda}$ as follows:

$$
\frac{\partial L(\mathbf{S})}{\partial\boldsymbol{\lambda}}=\left(\frac{\partial L(\mathbf{S})}{\partial\lambda_1}\frac{\partial L(\mathbf{S})}{\partial\lambda_2}\frac{\partial L(\mathbf{S})}{\partial\lambda_3}\frac{\partial L(\mathbf{S})}{\partial\lambda_4}\right)^{\mathrm{T}}=\mathbf{0}\in\mathbf{R}^4
\tag{16}
$$

The state equation is derived from the above equation. It consists of the continuum and the Navier–Stokes equations. The Navier–Stokes equation consists of the time derivative term, the advection term, the pressure term, and the diffusion term as follows:

$$
\frac{\partial u}{\partial x}+\frac{\partial v}{\partial y}+\frac{\partial w}{\partial z}=0\quad\text{in }\Omega
\tag{17}
$$

$$
-\frac{\partial u}{\partial t}-\frac{1}{\rho}\frac{\partial p}{\partial x}+2v\frac{\partial}{\partial x}\frac{\partial u}{\partial x}+v\frac{\partial}{\partial y}\left(\frac{\partial v}{\partial x}+\frac{\partial u}{\partial y}\right)+v\frac{\partial}{\partial z}\left(\frac{\partial w}{\partial x}+\frac{\partial u}{\partial z}\right)=0\quad\text{in }\Omega
\tag{18}
$$

$$
-\frac{\partial v}{\partial t}-\frac{1}{\rho}\frac{\partial p}{\partial y}+v\frac{\partial}{\partial x}\left(\frac{\partial u}{\partial y}+\frac{\partial v}{\partial x}\right)+2v\frac{\partial}{\partial y}\frac{\partial v}{\partial y}+v\frac{\partial}{\partial z}\left(\frac{\partial w}{\partial y}+\frac{\partial v}{\partial z}\right)=0\quad\text{in }\Omega
\tag{19}
$$

$$
-\frac{\partial w}{\partial t}-\frac{1}{\rho}\frac{\partial p}{\partial z}+v\frac{\partial}{\partial x}\left(\frac{\partial u}{\partial z}+\frac{\partial w}{\partial x}\right)+v\frac{\partial}{\partial y}\left(\frac{\partial v}{\partial z}+\frac{\partial w}{\partial y}\right)+2v\frac{\partial}{\partial z}\frac{\partial w}{\partial z}=0\quad\text{in }\Omega
\tag{20}
$$

In the state equation, the boundary condition is shown as Table I.

Table I. Boundary conditions.

Domain	State equations	Adjoint equations
Γ_w	$u=0.01, v=0, w=0$	$\lambda_1=0, \lambda_2=0, \lambda_3=0, \lambda_4=0$
Γ_E	$p=0$	$\lambda_1=0, \lambda_2=0, \lambda_3=0, \lambda_4=0$
Γ_N, Γ_S	$v=0$	$\lambda_1=0, \lambda_2=0, \lambda_3=0, \lambda_4=0$
Γ_U, Γ_D	$w=0$	$\lambda_1=0, \lambda_2=0, \lambda_3=0, \lambda_4=0$
γ	$u=0, v=0, w=0$	$\lambda_2=1, \lambda_3=0, \lambda_4=0$

DOI: 10.1002/fld

2.4. Adjoint equations

The adjoint variable is calculated by solving the adjoint equation, the stationary condition that is obtained by taking the derivative of the Lagrange function with respect to the state variable **W** as follows:

$$\frac{\partial L(\mathbf{S})}{\partial \mathbf{W}} = \left(\frac{\partial L(\mathbf{S})}{\partial p} \frac{\partial L(\mathbf{S})}{\partial u} \frac{\partial L(\mathbf{S})}{\partial v} \frac{\partial L(\mathbf{S})}{\partial w} \right)^{\mathrm{T}} = \mathbf{0} \in \mathbf{R}^4 \tag{21}$$

The equation derived from the first element of Equation (21) is as follows:

$$\begin{aligned} \frac{\partial L(\mathbf{S})}{\partial p} = & \int_{t_{\mathrm{s}}}^{t_{\mathrm{e}}} \int_{\gamma(\mathbf{S})} \frac{1}{\rho} n_1 \, \mathrm{d}\gamma \, \mathrm{d}t + \int_{t_{\mathrm{s}}}^{t_{\mathrm{e}}} \int_{\Gamma(\mathbf{S})} \frac{1}{\rho} (-\lambda_{i+1} n_i) \, \mathrm{d}\Gamma \, \mathrm{d}t \\ & + \int_{t_{\mathrm{s}}}^{t_{\mathrm{e}}} \int_{\Omega(\mathbf{S})} \frac{1}{\rho} \left(\frac{\partial \lambda_2}{\partial x} + \frac{\partial \lambda_3}{\partial y} + \frac{\partial \lambda_4}{\partial z} \right) \mathrm{d}\Omega \, \mathrm{d}t = 0 \in \mathbf{R}^1 \end{aligned} \tag{22}$$

Since the above equation should be satisfied at arbitrary time and space, the computational domain Ω term, the boundary Γ term, and the surface boundary γ on the object should be equal to zero. Thus, from Ω term of Equation (22), we obtain

$$\frac{\partial \lambda_2}{\partial x} + \frac{\partial \lambda_3}{\partial y} + \frac{\partial \lambda_4}{\partial z} = 0 \quad \text{on } \Omega \tag{23}$$

γ term is as follows:

$$n_1 - \lambda_2 n_1 - \lambda_3 n_2 - \lambda_4 n_3 = 0 \quad \text{on } \gamma \tag{24}$$

The above equation should be satisfied at arbitrary normal vector as follows:

$$(\lambda_2 \;\; \lambda_3 \;\; \lambda_4) = (1 \;\; 0 \;\; 0) \quad \text{on } \gamma \tag{25}$$

The ψ term is as follows:

$$-\lambda_2 n_1 - \lambda_3 n_2 - \lambda_4 n_3 = 0 \quad \text{on } \psi \tag{26}$$

The above equation also should be satisfied at the arbitrary normal vector as follows:

$$(\lambda_2 \;\; \lambda_3 \;\; \lambda_4) = (0 \;\; 0 \;\; 0) \quad \text{on } \psi \tag{27}$$

The other boundary condition is determined in a similar way (see Table I). The equation derived from the second element of Equation (15) is as follows:

$$\begin{aligned} \frac{\partial L(\mathbf{S})}{\partial u} = & -\int_{t_{\mathrm{s}}}^{t_{\mathrm{e}}} \int_{\Gamma(\mathbf{S})} \left\{ -\frac{1}{\rho} \lambda_1 n_1 + v \left(\frac{\partial \lambda_2}{\partial x_j} + \frac{\partial \lambda_{j+1}}{\partial x} \right) n_j \right\} \mathrm{d}\Gamma \, \mathrm{d}t \\ & + \int_{t_{\mathrm{s}}}^{t_{\mathrm{e}}} \int_{\Omega(\mathbf{S})} \left\{ \frac{\partial \lambda_2}{\partial t} - \frac{1}{\rho} \frac{\partial \lambda_1}{\partial x} + v \frac{\partial}{\partial x_j} \left(\frac{\partial \lambda_{j+1}}{\partial x} + \frac{\partial \lambda_2}{\partial x_j} \right) \right\} \mathrm{d}\Omega \, \mathrm{d}t \\ & - \int_{\Omega(\mathbf{S})} [\lambda_2]_{t_s}^{t_e} \, \mathrm{d}\Omega = 0 \in \mathbf{R}^1 \end{aligned} \tag{28}$$

The first integral term of the right side is as follows:

$$-\lambda_1 n_1+\mu\left(\frac{\partial\lambda_2}{\partial x}+\frac{\partial\lambda_2}{\partial x}\right)n_1+\mu\left(\frac{\partial\lambda_2}{\partial y}+\frac{\partial\lambda_3}{\partial x}\right)n_2+\mu\left(\frac{\partial\lambda_2}{\partial z}+\frac{\partial\lambda_4}{\partial x}\right)n_3=0\in\mathbf{R}^1 \quad \text{on } \Gamma \tag{29}$$

The third integral term of the right side is as follows:

$$\lambda_2(t_\mathrm{e},\mathbf{X})-\lambda_2(t_\mathrm{s},\mathbf{X})=0\in\mathbf{R}^1 \quad \text{in } \Omega \tag{30}$$

The second integral term of the right side is as follows:

$$\frac{\partial\lambda_2}{\partial t}-\frac{1}{\rho}\frac{\partial\lambda_1}{\partial x}+2\nu\frac{\partial}{\partial x}\frac{\partial\lambda_2}{\partial x}+\nu\frac{\partial}{\partial y}\left(\frac{\partial\lambda_2}{\partial y}+\frac{\partial\lambda_3}{\partial x}\right)+\nu\frac{\partial}{\partial z}\left(\frac{\partial\lambda_2}{\partial z}+\frac{\partial\lambda_4}{\partial x}\right)=0\in\mathbf{R}^1 \quad \text{in } \Omega \tag{31}$$

Equation (31) causes inverse diffusion problems because the viscosity term in Equation (30) has the same positive sign as the time derivative term [29, 30]. Inverse diffusion problems cause numerical oscillations and cannot converge. For stability reasons [31], the backward time is defined as follows:

$$t=-\tau\in\mathbf{R}^1 \quad \text{in } \Omega \tag{32}$$

By using Equation (32), Equation (31) becomes as follows:

$$-\frac{\partial\lambda_2}{\partial \tau}-\frac{1}{\rho}\frac{\partial\lambda_1}{\partial x}+2\nu\frac{\partial}{\partial x}\frac{\partial\lambda_2}{\partial x}+\nu\frac{\partial}{\partial y}\left(\frac{\partial\lambda_2}{\partial y}+\frac{\partial\lambda_3}{\partial x}\right)+\nu\frac{\partial}{\partial z}\left(\frac{\partial\lambda_2}{\partial z}+\frac{\partial\lambda_4}{\partial x}\right)=0\in\mathbf{R}^1 \quad \text{in } \Omega \tag{33}$$

Adjoint equations are eventually calculated by using stokes equations in backward time, while the state equations are calculated by using Stokes equation in forward time.

Similarly, the equations derived from the third element and the forth element are as follows:

$$-\lambda_1 n_2+\mu\left(\frac{\partial\lambda_2}{\partial y}+\frac{\partial\lambda_3}{\partial x}\right)n_1+\mu\left(\frac{\partial\lambda_3}{\partial y}+\frac{\partial\lambda_3}{\partial y}\right)n_2+\mu\left(\frac{\partial\lambda_3}{\partial z}+\frac{\partial\lambda_4}{\partial y}\right)n_3=0\in\mathbf{R}^1 \quad \text{on } \Gamma \tag{34}$$

$$\lambda_3(t_\mathrm{e},\mathbf{X})-\lambda_3(t_\mathrm{s},\mathbf{X})=0\in\mathbf{R}^1 \quad \text{in } \Omega \tag{35}$$

$$-\frac{\partial\lambda_3}{\partial \tau}-\frac{1}{\rho}\frac{\partial\lambda_1}{\partial y}+\nu\frac{\partial}{\partial x}\left(\frac{\partial\lambda_2}{\partial y}+\frac{\partial\lambda_3}{\partial x}\right)+\nu\frac{\partial}{\partial y}\left(\frac{\partial\lambda_3}{\partial y}+\frac{\partial\lambda_3}{\partial y}\right)+\nu\frac{\partial}{\partial z}\left(\frac{\partial\lambda_3}{\partial z}+\frac{\partial\lambda_4}{\partial y}\right)=0\in\mathbf{R}^1 \quad \text{in } \Omega \tag{36}$$

$$-\lambda_1 n_3+\mu\left(\frac{\partial\lambda_2}{\partial z}+\frac{\partial\lambda_4}{\partial x}\right)n_1+\mu\left(\frac{\partial\lambda_4}{\partial y}+\frac{\partial\lambda_3}{\partial z}\right)n_2+\mu\left(\frac{\partial\lambda_4}{\partial z}+\frac{\partial\lambda_4}{\partial z}\right)n_3=0\in\mathbf{R}^1 \quad \text{on } \Gamma \tag{37}$$

$$\lambda_4(t_\mathrm{e},\mathbf{X})-\lambda_4(t_\mathrm{s},\mathbf{X})=0\in\mathbf{R}^1 \quad \text{in } \Omega \tag{38}$$

$$-\frac{\partial\lambda_4}{\partial \tau}-\frac{1}{\rho}\frac{\partial\lambda_1}{\partial z}+\nu\frac{\partial}{\partial x}\left(\frac{\partial\lambda_2}{\partial z}+\frac{\partial\lambda_4}{\partial x}\right)+\nu\frac{\partial}{\partial y}\left(\frac{\partial\lambda_4}{\partial y}+\frac{\partial\lambda_3}{\partial z}\right)+\nu\frac{\partial}{\partial z}\left(\frac{\partial\lambda_4}{\partial z}+\frac{\partial\lambda_4}{\partial z}\right)=0\in\mathbf{R}^1 \quad \text{in } \Omega \tag{39}$$

2.5. Sensitivity equations

The derivative of the Lagrange function with respect to $\mathbf{X}\in\mathbf{R}^3$ is the sensitivity equation. The value of the partial differential represents the sensitivity. The sensitivity equation is as follows:

$$\frac{\partial L(\mathbf{S})}{\partial \mathbf{X}}=\left(\frac{\partial L(\mathbf{S})}{\partial x}\frac{\partial L(\mathbf{S})}{\partial y}\frac{\partial L(\mathbf{S})}{\partial z}\right)^\mathrm{T}=\mathbf{0} \tag{40}$$

DOI: 10.1002/fld

The equation derived from the first element of Equation (40) is as follows:

$$\begin{aligned}
&\left\{-\lambda_1 n_1+\mu\left(\frac{\partial\lambda_2}{\partial x}+\frac{\partial\lambda_2}{\partial x}\right)n_1+\mu\left(\frac{\partial\lambda_2}{\partial y}+\frac{\partial\lambda_3}{\partial x}\right)n_2+\mu\left(\frac{\partial\lambda_2}{\partial z}+\frac{\partial\lambda_4}{\partial x}\right)n_3\right\}\frac{\partial u}{\partial x}\\
&\quad+\left\{-\lambda_1 n_2+\mu\left(\frac{\partial\lambda_2}{\partial y}+\frac{\partial\lambda_3}{\partial x}\right)n_1+\mu\left(\frac{\partial\lambda_3}{\partial y}+\frac{\partial\lambda_3}{\partial y}\right)n_2+\mu\left(\frac{\partial\lambda_3}{\partial z}+\frac{\partial\lambda_4}{\partial y}\right)n_3\right\}\frac{\partial v}{\partial x}\\
&\quad+\left\{-\lambda_1 n_3+\mu\left(\frac{\partial\lambda_2}{\partial z}+\frac{\partial\lambda_4}{\partial x}\right)n_1+\mu\left(\frac{\partial\lambda_4}{\partial y}+\frac{\partial\lambda_3}{\partial z}\right)n_2+\mu\left(\frac{\partial\lambda_4}{\partial z}+\frac{\partial\lambda_4}{\partial z}\right)n_3\right\}\frac{\partial w}{\partial x}\\
&=0\quad(=\delta S_1)\in\mathbf{R}^1\quad\text{on }\gamma
\end{aligned}\tag{41}$$

(See Appendix B for details regarding this transformation.)

The equation derived from the second element and third element in Equation (40) is as follows:

$$\begin{aligned}
&\left\{-\lambda_1 n_1+\mu\left(\frac{\partial\lambda_2}{\partial x}+\frac{\partial\lambda_2}{\partial x}\right)n_1+\mu\left(\frac{\partial\lambda_2}{\partial y}+\frac{\partial\lambda_3}{\partial x}\right)n_2+\mu\left(\frac{\partial\lambda_2}{\partial z}+\frac{\partial\lambda_4}{\partial x}\right)n_3\right\}\frac{\partial u}{\partial y}\\
&\quad+\left\{-\lambda_1 n_2+\mu\left(\frac{\partial\lambda_2}{\partial y}+\frac{\partial\lambda_3}{\partial x}\right)n_1+\mu\left(\frac{\partial\lambda_3}{\partial y}+\frac{\partial\lambda_3}{\partial y}\right)n_2+\mu\left(\frac{\partial\lambda_3}{\partial z}+\frac{\partial\lambda_4}{\partial y}\right)n_3\right\}\frac{\partial v}{\partial y}\\
&\quad+\left\{-\lambda_1 n_3+\mu\left(\frac{\partial\lambda_2}{\partial z}+\frac{\partial\lambda_4}{\partial x}\right)n_1+\mu\left(\frac{\partial\lambda_4}{\partial y}+\frac{\partial\lambda_3}{\partial z}\right)n_2+\mu\left(\frac{\partial\lambda_4}{\partial z}+\frac{\partial\lambda_4}{\partial z}\right)n_3\right\}\frac{\partial w}{\partial y}\\
&=0\quad(=\delta S_2)\in\mathbf{R}^1\quad\text{on }\gamma
\end{aligned}\tag{42}$$

$$\begin{aligned}
&\left\{-\lambda_1 n_1+\mu\left(\frac{\partial\lambda_2}{\partial x}+\frac{\partial\lambda_2}{\partial x}\right)n_1+\mu\left(\frac{\partial\lambda_2}{\partial y}+\frac{\partial\lambda_3}{\partial x}\right)n_2+\mu\left(\frac{\partial\lambda_2}{\partial z}+\frac{\partial\lambda_4}{\partial x}\right)n_3\right\}\frac{\partial u}{\partial z}\\
&\quad+\left\{-\lambda_1 n_2+\mu\left(\frac{\partial\lambda_2}{\partial y}+\frac{\partial\lambda_3}{\partial x}\right)n_1+\mu\left(\frac{\partial\lambda_3}{\partial y}+\frac{\partial\lambda_3}{\partial y}\right)n_2+\mu\left(\frac{\partial\lambda_3}{\partial z}+\frac{\partial\lambda_4}{\partial y}\right)n_3\right\}\frac{\partial v}{\partial z}\\
&\quad+\left\{-\lambda_1 n_3+\mu\left(\frac{\partial\lambda_2}{\partial z}+\frac{\partial\lambda_4}{\partial x}\right)n_1+\mu\left(\frac{\partial\lambda_4}{\partial y}+\frac{\partial\lambda_3}{\partial z}\right)n_2+\mu\left(\frac{\partial\lambda_4}{\partial z}+\frac{\partial\lambda_4}{\partial z}\right)n_3\right\}\frac{\partial w}{\partial z}\\
&=0\quad(=\delta S_3)\in\mathbf{R}^1\quad\text{on }\gamma
\end{aligned}\tag{43}$$

2.6. Shape modification

The variables δS_1–δS_3 of Equations (41)–(43) make the sensitivity. When the surface $\mathbf{S}$ takes extremal values, this sensitivity equals zero. By using the steepest decent method, the shape is modified so that the sensitivity becomes zero:

$$\mathbf{S}_{(k+1)}=\mathbf{S}_{(k)}+\beta\int_{t_\mathrm{s}}^{t_\mathrm{e}}\boldsymbol{\delta}\mathbf{S}_{(k)}\,\mathrm{d}t,\quad k=0,1,\ldots\in\mathbf{R}^3\quad\text{on }\gamma\tag{44}$$

The value of the coefficient β should be small enough in order to robustly converge to the optimal coordinates and to avoid collapse of the mesh topology. The value is decided based on a heuristic search method [32].

3. SMOOTHING

In shape optimization, meshes have been mainly analyzed by using 2D low-resolution models. Shapes were optimized by controlling the nodes located on the surface. This approach is called the mesh point approach [33]. Unfortunately, in the case of 3D high-resolution unstructured grid (especially tetrahedral element), the mesh point approach does not work so well. As a node on the boundary in a finite element is moved along the sensitivity, the smooth shape is lost and an irregular shape is constructed [27]. As the surface becomes irregular, it partly causes a numerical vibration in fluid analysis and negative volumes of mesh elements. In this study, Laplace smoothing method is applied to the surface shape, as this method can be easily implemented [34–36].

$$\delta\mathbf{S}_{[i]} = \frac{\sum_j \delta\overline{\mathbf{S}}_{\langle j\rangle} A_{\langle j\rangle}}{\sum_j A_{\langle j\rangle}} \quad \text{on } \gamma \tag{45}$$

Laplace smoothing method is one of the methods where the averaged movement amount of an element is converted to the movement amount of a node. The variable $\delta\mathbf{S}_{[i]}$ shows the movement amount of a node point i (in Figure 3). The variable $\delta\overline{\mathbf{S}}_{\langle j\rangle}$ represents the movement amount at gravity position of element j. The variable $A_{\langle j\rangle}$ stands for an area of an element j. By updating the movement amount iteratively, the deformed surface mesh is constructed. We noted the following:

Point A: In the case that some elements turned inside out when a node i moved to a new node position, this node does not execute the move.

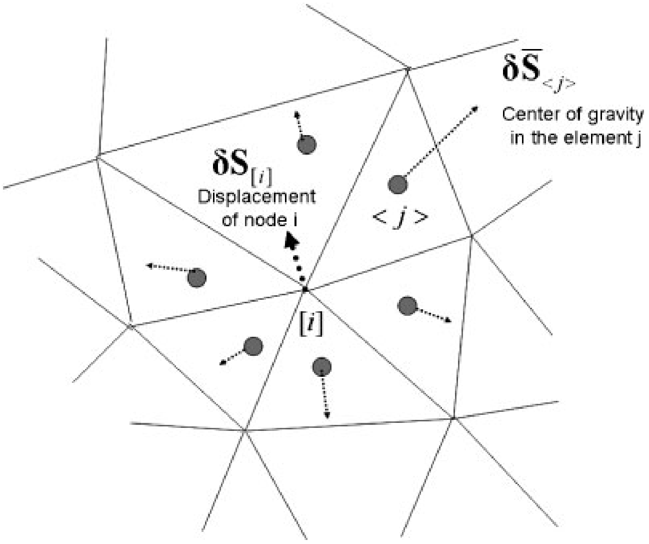

$\delta\overline{\mathbf{S}}_{<j>}$
Center of gravity in the element j
$\delta\mathbf{S}_{[i]}$
Displacement of node i
< j >
[i]

Figure 3. Smoothing on the surface of the object.

DOI: 10.1002/fld

Point B: The Laplace smoothing method is sequential. The number of necessary iterations cannot be known in advance. If the number of iterations is relatively large, the sharp edges will be smoothed out. On the contrary, if only a small number of iterations are applied, the awkward shape of the surface will make the optimization calculations impossible. In this study, we empirically choose to apply three smoothing iterations.

4. CONSTANT VOLUME CONSTRAINT

By using only Equations (41)–(43), due to the volume becoming negative, the object may cause an unrealistic deformation. This problem can be overcome by considering constraints. In this research, a constant volume constraint is implemented. Two approaches are considered.

Process A: By introducing a new adjoint variable, the constant volume constraint is added to the Lagrange function. The sensitivity is modified to satisfy the constant volume constraint.

Process B: After constructing the optimal shape with respect to the cost function, this shape is iteratively modified to satisfy the constant volume constraint.

In the former approach, the sensitivity at every shape step includes the constant volume constraint. The constant volume condition could not be sufficiently satisfied by this approach [37]. In this study, the latter approach is implemented. Line search method is applied to the constant volume constraint. The function $h(\mathbf{S})$ is defined as follows:

$$h(\mathbf{S}_{(k)}) = V(\mathbf{S}_{(k)}) - V(\mathbf{S}_{(0)}), \quad k = 0, 1, \ldots \in \mathbf{R}^1 \tag{46}$$

Minimizing the function $h(\mathbf{S})$ means satisfying the constant volume constraint. To minimize $h(\mathbf{S})$ with maintaining the surface shape, the surface shape is deformed along an outward normal vector

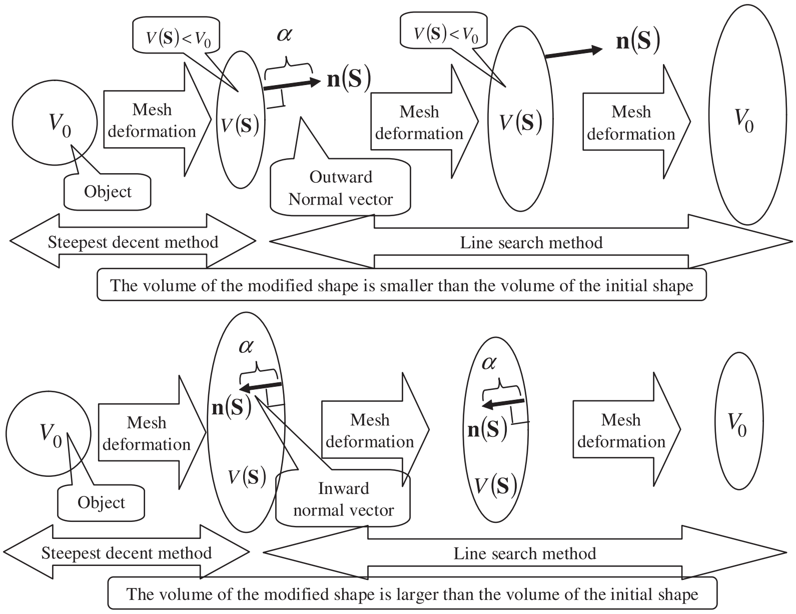


Figure 4. Volume constant.

DOI: 10.1002/fld

of the object as follows:

$$\mathbf{S}_{(k),(l+1)} = \mathbf{S}_{(k),(l)} + \alpha h(\mathbf{S}_{(k),(l)})\mathbf{n}(\mathbf{S}_{(k)}), \quad l = 0, 1, \ldots \in \mathbf{R}^3 \quad \text{on } \gamma \tag{47}$$

The subscript (l) is the iteration number of the mesh deformation. As (l) is increased, the volume of the deformed shape gets closer to the volume of the initial shape. $\mathbf{n}(\mathbf{S}_{(k)})$ remains constant while the subscript (l) is increased. The deformation amount is set to be small by multiplying a coefficient α in the second term. The volume constant mechanism is shown in Figure 4. In the beginning, the shape is deformed using sensitivity analysis based on the adjoint variable method. In case the deformed volume is smaller than the initial volume, the volume is slowly increased along an outward normal vector of the object surface by expanding the shape. In case that the deformed volume is larger than the initial volume, the deformed volume is slowly decreased along an inward normal vector of the object surface by suppressing the shape. In other words, this algorithm is repeated until the deformed shape is in good agreement with the initial volume.

5. ROBUST MESH DEFORMATION

In the field of shape optimization, the mesh is mainly deformed by using the Laplace equation, as the deformation of shape is rather small. The Laplace equation is as follows:

$$\nabla^2 \Theta(\mathbf{X}) = 0 \in \mathbf{R}^3 \quad \text{in } \Omega \tag{48}$$

$$\Theta(\mathbf{X}) = \beta \int_{t_s}^{t_e} \boldsymbol{\delta}\mathbf{S}_{(k)}\, \mathrm{d}t \in \mathbf{R}^3 \quad \text{on } \gamma \tag{49}$$

$$\Theta(\mathbf{X}) = 0 \in \mathbf{R}^3 \quad \text{on } \Gamma_\mathrm{W}, \Gamma_\mathrm{N}, \Gamma_\mathrm{E}, \Gamma_\mathrm{S} \tag{50}$$

In this approach, the object's surface is deformed to satisfy the Dirichlet boundary condition. However, some elements might have partially negative volumes, causing the calculation to fail. To robustly relocate nodes, the biharmonic equation consisting of the fourth derivative is applied to deforming the mesh [38]

$$\nabla^4 \Theta(\mathbf{X}) = 0 \in \mathbf{R}^3 \quad \text{in } \Omega \tag{51}$$

$$\Theta(\mathbf{X}) = \beta \int_{t_s}^{t_e} \boldsymbol{\delta}\mathbf{S}_{(k)}\, \mathrm{d}t \in \mathbf{R}^3 \quad \text{on } \gamma \tag{52}$$

$$\Theta(\mathbf{X}) = 0 \in \mathbf{R}^3 \quad \text{on } \Gamma_\mathrm{W}, \Gamma_\mathrm{N}, \Gamma_\mathrm{E}, \Gamma_\mathrm{S} \tag{53}$$

6. DISCRETIZATION

This section describes the discretization procedure used in the present study. In the shape optimization of an object located in a flow, the finite difference method and the finite volume method have been generally employed since the 1980s. In this study, we use the FEM to discretize the adjoint equation, sensitivity equation, and biharmonic equation.

6.1. Shape function

Figure 5 illustrates a tetrahedral element, consisting of nodes $i, j, k,$ and l. The shape function in the element is defined as follows:

$$N_{[i]} = \frac{a_{[i]} + b_{[i]}x + c_{[i]}y + d_{[i]}z}{6V}, \quad i = 1, 2, 3, 4 \tag{54}$$

with

$$6V = \det \begin{vmatrix} 1 & x_{[i]} & y_{[i]} & z_{[i]} \\ 1 & x_{[j]} & y_{[j]} & z_{[j]} \\ 1 & x_{[k]} & y_{[k]} & z_{[k]} \\ 1 & x_{[l]} & y_{[l]} & z_{[l]} \end{vmatrix} \tag{55}$$

where V represents the volume of the tetrahedron. By expanding the other relevant determinants into their cofactors, we have

$$\begin{gathered} a_{[i]} = \det \begin{vmatrix} x_{[j]} & y_{[j]} & z_{[j]} \\ x_{[k]} & y_{[k]} & z_{[k]} \\ x_{[l]} & y_{[l]} & z_{[l]} \end{vmatrix}, \quad b_{[i]} = -\det \begin{vmatrix} 1 & y_{[j]} & z_{[j]} \\ 1 & y_{[k]} & z_{[k]} \\ 1 & y_{[l]} & z_{[l]} \end{vmatrix} \\ c_{[i]} = -\det \begin{vmatrix} x_{[j]} & 1 & z_{[j]} \\ x_{[k]} & 1 & z_{[k]} \\ x_{[l]} & 1 & z_{[l]} \end{vmatrix}, \quad d_{[i]} = -\det \begin{vmatrix} x_{[j]} & y_{[j]} & 1 \\ x_{[k]} & y_{[k]} & 1 \\ x_{[l]} & y_{[l]} & 1 \end{vmatrix}, \quad i, j, k, l = 1, 2, 3, 4 \end{gathered} \tag{56}$$

with the other constants obtained by cyclic interchange of the subscripts in the order i, j, k, l. By using a determinant form, we can obtain unknown variables as follows:

$$u_{\langle m \rangle} = N_{[i]}u_{[i]} + N_{[j]}u_{[j]} + N_{[k]}u_{[k]} + N_{[l]}u_{[l]} \tag{57}$$

6.2. Discretization of the adjoint equation

In this study, the adjoint equation is solved by using the fractional step method. At first, the time derivative in the adjoint equation is discretized as follows:

$$\frac{\lambda_{i+1}^{(n+1)} - \lambda_{i+1}^{(n)}}{\Delta t} = -\frac{1}{\rho}\frac{\partial \lambda_1^{(n+1)}}{\partial x_i} + v\frac{\partial}{\partial x_j}\left(\frac{\partial \lambda_{i+1}^{(n)}}{\partial x_j} + \frac{\partial \lambda_{j+1}^{(n)}}{\partial x_i}\right), \quad i, j = 1, 2, 3 \tag{58}$$

where Δt represents the time step. For the convenience of the following discussion, an intermediate velocity $\widetilde{\lambda}_{i+1}$ $(i = 1, 2, 3)$ is defined as follows:

$$\widetilde{\lambda}_{i+1} = \lambda_{i+1}^{(n)} + \Delta t v \frac{\partial}{\partial x_j}\left(\frac{\partial \lambda_{i+1}^{(n)}}{\partial x_j} + \frac{\partial \lambda_{j+1}^{(n)}}{\partial x_i}\right), \quad i, j = 1, 2, 3 \tag{59}$$

DOI: 10.1002/fld

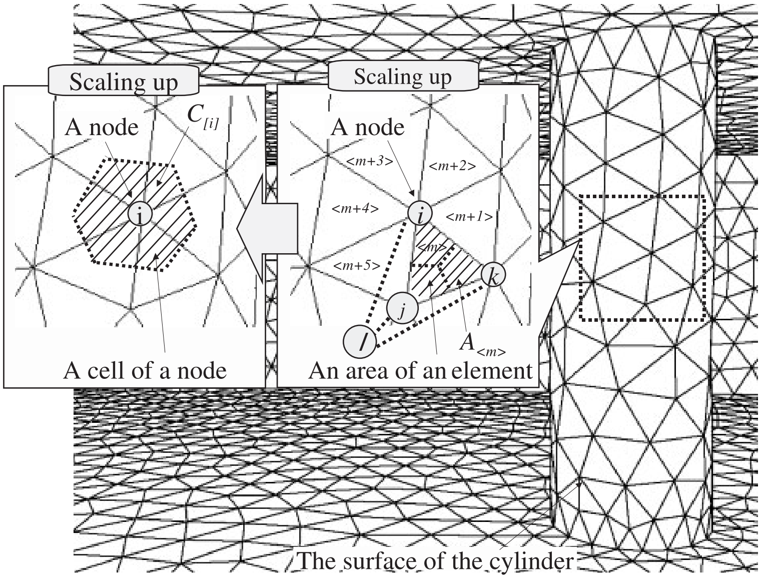


Figure 5. An element (four node tetrahedron) in the mesh.

Applying the divergence to Equation (58) and using Equations (23) and (59), a Poisson equation for the adjoint pressure is derived as follows:

$$\frac{\partial^2 \lambda_1^{(n+1)}}{\partial x^2}+\frac{\partial^2 \lambda_1^{(n+1)}}{\partial y^2}+\frac{\partial^2 \lambda_1^{(n+1)}}{\partial z^2}=\frac{\rho}{\Delta t}\left(\frac{\partial \widetilde{\lambda}_2}{\partial x}+\frac{\partial \widetilde{\lambda}_3}{\partial y}+\frac{\partial \widetilde{\lambda}_4}{\partial z}\right), \quad i,j=1,2,3 \tag{60}$$

Substituting Equation (59) into Equation (58), an equation for updating the adjoint velocity vector can be obtained as follows:

$$\lambda_{i+1}^{(n+1)}=\widetilde{\lambda}_{i+1}-\frac{\Delta t}{\rho}\frac{\partial \lambda_1^{(n+1)}}{\partial x_i}, \quad i=1,2,3 \tag{61}$$

By using the weight function ($\lambda_i^*,\ i=1,2,3,4$), the residual equations are derived as follows:

$$\int_\Omega \lambda_{i+1}^* \widetilde{\lambda}_{i+1}\,\mathrm{d}\Omega=\int_\Omega \lambda_{i+1}^*\lambda_{i+1}^{(n)}\,\mathrm{d}\Omega-\Delta t v\int_\Omega \frac{\partial \lambda_{i+1}^*}{\partial x_j}\left(\frac{\partial \lambda_{i+1}^{(n)}}{\partial x_j}+\frac{\partial \lambda_{j+1}^{(n)}}{\partial x_i}\right)\mathrm{d}\Omega, \quad i,j=1,2,3 \tag{62}$$

$$\int_\Omega \frac{\partial \lambda_1^*}{\partial x_i}\frac{\partial \lambda_1^{(n+1)}}{\partial x_i}\,\mathrm{d}\Omega=-\frac{\rho}{\Delta t}\int_\Omega \lambda_1^*\left(\frac{\partial \widetilde{\lambda}_2}{\partial x}+\frac{\partial \widetilde{\lambda}_3}{\partial y}+\frac{\partial \widetilde{\lambda}_4}{\partial z}\right)\mathrm{d}\Omega, \quad i=1,2,3 \tag{63}$$

$$\int_\Omega \lambda_{i+1}^*\lambda_{i+1}^{(n+1)}\,\mathrm{d}\Omega=\int_\Omega \lambda_{i+1}^*\widetilde{\lambda}_{i+1}\,\mathrm{d}\Omega-\frac{\Delta t}{\rho}\int_\Omega \lambda_{i+1}^*\frac{\partial \lambda_1^{(n+1)}}{\partial x_i}\,\mathrm{d}\Omega, \quad i=1,2,3 \tag{64}$$

where the Gauss–Green theorem was used to derive these equations. The integral term with respect to Γ is zero because the natural boundary condition is satisfied. The form of the weight function is assumed to be the same form as the unknown function. Upon substitution of Equation (57), the residual equations (62)–(64) are transformed as follows:

$$\begin{aligned}
&\int_\Omega N_{[\alpha]} N_{[\beta]}\,\mathrm{d}\Omega \widetilde{\lambda}_{i+1,[\beta]}\\
&\quad=\int_\Omega N_{[\alpha]} N_{[\beta]}\,\mathrm{d}\Omega \lambda^{(n)}_{i+1,[\beta]}\\
&\quad\quad-\Delta t v\left\{\int_\Omega \left(\frac{\partial N_{[\alpha]}}{\partial x_j}\frac{\partial N_{[\beta]}}{\partial x_j}\right)\mathrm{d}\Omega \lambda^{(n)}_{i+1,[\beta]}+\int_\Omega \left(\frac{\partial N_{[\alpha]}}{\partial x_j}\frac{\partial N_{[\beta]}}{\partial x_i}\right)\mathrm{d}\Omega \lambda^{(n)}_{j+1,[\beta]}\right\}\\
&\quad\quad i,j=1,2,3,\quad \alpha,\beta=1,2,3,4
\end{aligned} \tag{65}$$

$$\int_\Omega \frac{\partial N_{[\alpha]}}{\partial x_i}\frac{\partial N_{[\beta]}}{\partial x_i}\,\mathrm{d}\Omega \lambda^{(n+1)}_{1,[\beta]}=-\frac{\rho}{\Delta t}\int_\Omega N_{[\alpha]}\frac{\partial N_{[\beta]}}{\partial x_i}\,\mathrm{d}\Omega \widetilde{\lambda}_{j,[\beta]},\quad i,j=1,2,3,\ \ \alpha,\beta=1,2,3,4 \tag{66}$$

$$\begin{aligned}
&\int_\Omega N_{[\alpha]} N_{[\beta]}\,\mathrm{d}\Omega \lambda^{(n+1)}_{i+1,[\beta]}=\int_\Omega N_{[\alpha]} N_{[\beta]}\,\mathrm{d}\Omega \lambda^{(n+1)}_{i+1,[\beta]}-\frac{\Delta t}{\rho}\int_\Omega N_{[\alpha]}\frac{\partial N_{[\beta]}}{\partial x_i}\,\mathrm{d}\Omega \lambda^{(n+1)}_{1}\\
&i=1,2,3,\ \ \alpha,\beta=1,2,3,4
\end{aligned} \tag{67}$$

The local matrices in the finite element equations (Equations (65)–(67)) are converted to a global matrix, which is solved by using the HEC-MW solver [28]. We take the same approach for the state equations.

6.3. Discretization of sensitivity equation

In this section, the algorithm to obtain the sensitivity is shown in a 9-step procedure.

Step 1: According to the HEC-MW text format, the connectivity with respect to the boundary γ is made [28]. The mesh data structure contains a list of nodes and a list of elements. The former contains nodal coordinates, while the latter contains node indices, in order to describe the connectivity of the mesh.

Step 2: Both an element (four nodes i, j, k, l) and a surface are selected by using the connectivity data.

Step 3: The coefficients a, b, c, d (Equation (56)) are calculated.

Step 4: Partial differentiations in the sensitivity equations are computed. The partial differentiation of u with respect to x, for example, is determined as follows:

$$\left(\frac{\partial u}{\partial x}\right)_{\langle m\rangle}=b_{[i]}u_{[i]}+b_{[j]}u_{[j]}+b_{[k]}u_{[k]}+b_{[l]}u_{[l]}=b_{[l]}u_{[l]} \tag{68}$$

where $\langle m\rangle$ shows the element number and $[i], [j], [k], [l]$ show the node numbers. The state variable u has been already obtained using the state equations. As shown in Figure 5, the nodes $[i], [j], [k]$ are on the surface γ and the node $[l]$ is in the space. By using the boundary condition in Table I, $u_{[i]}, u_{[j]}, u_{[k]}$ become zero. Equation (68) is the value of the element $\langle m\rangle$. The value of the element $\langle m\rangle$ needs to be converted to a value of a node to calculate the sensitivity.

Step 5: The partial differentiation value at an element is divided into the three node values (i, j, k) in proportion to the surface area. In the data structure of HEC-MW, the value (sensitivity equation) of the node and the cell area as shown in Figure 5 cannot be obtained directly, while the value of the node can be partly calculated by using the value of the element. The node i, for example, is

DOI: 10.1002/fld

connected to six elements as shown in Figure 5. The sixth part of the value corresponding to the node i, j, k can be obtained as follows:

$$\left(\frac{\partial u}{\partial x}\right)_{\langle m\rangle}\frac{A_{\langle m\rangle}}{3} \tag{69}$$

where $A_{\langle m\rangle}$ indicates the surface area in element $\langle m\rangle$. If the area of element $\langle m\rangle$ is large, the value of the partial differentiation in element $\langle m\rangle$ is dominant at node $[i]$.

Step 6: The area in element $\langle m\rangle$ is divided into three cells:

$$\frac{A_{\langle m\rangle}}{3} \tag{70}$$

Step 7: Steps 2–6 are repeated for all surfaces.

Step 8: The partial differentiation of u with respect to x is obtained as follows:

$$\begin{aligned}\left(\frac{\partial u}{\partial x}\right)_{[i]}A_{[i]} &= \left(\frac{\partial u}{\partial x}\right)_{\langle m\rangle}\frac{A_{\langle m\rangle}}{3}+\left(\frac{\partial u}{\partial x}\right)_{\langle m+1\rangle}\frac{A_{\langle m+1\rangle}}{3}+\left(\frac{\partial u}{\partial x}\right)_{\langle m+2\rangle}\frac{A_{\langle m+2\rangle}}{3}\\ &\quad+\left(\frac{\partial u}{\partial x}\right)_{\langle m+2\rangle}\frac{A_{\langle m+3\rangle}}{3}+\left(\frac{\partial u}{\partial x}\right)_{\langle m+4\rangle}\frac{A_{\langle m+4\rangle}}{3}+\left(\frac{\partial u}{\partial x}\right)_{\langle m+5\rangle}\frac{A_{\langle m+5\rangle}}{3}\end{aligned} \tag{71}$$

where the value of the node i can be obtained by summing parts of the value after finishing step 7. The other terms in the sensitivity equations are obtained in a similar way.

Step 9: The sensitivity $\delta\mathbf{S}$ at every node is calculated. δS_1, for example, is determined as follows:

$$\begin{aligned}\delta S_{1,[i]} &= \left\{-\lambda_{1,[i]}n_{1,[i]}+2\mu\left(\frac{\partial\lambda_2}{\partial x}\right)_{[i]}n_{1,[i]}+\mu\left\{\left(\frac{\partial\lambda_2}{\partial y}\right)_{[i]}+\left(\frac{\partial\lambda_3}{\partial x}\right)_{[i]}\right\}n_{2,[i]}\right.\\ &\quad\left.+\mu\left\{\left(\frac{\partial\lambda_2}{\partial z}\right)_{[i]}+\left(\frac{\partial\lambda_4}{\partial x}\right)_{[i]}\right\}n_{3,[i]}\right\}\left(\frac{\partial u}{\partial x}\right)_{[i]}\\ &\quad+\left\{-\lambda_{1,[i]}n_{2,[i]}+\mu\left\{\left(\frac{\partial\lambda_2}{\partial y}\right)_{[i]}+\left(\frac{\partial\lambda_3}{\partial x}\right)_{[i]}\right\}n_{1,[i]}+2\mu\left(\frac{\partial\lambda_3}{\partial y}\right)_{[i]}n_{2,[i]}\right.\\ &\quad\left.+\mu\left\{\left(\frac{\partial\lambda_3}{\partial z}\right)_{[i]}+\left(\frac{\partial\lambda_4}{\partial y}\right)_{[i]}\right\}n_{3,[i]}\right\}\left(\frac{\partial v}{\partial x}\right)_{[i]}\\ &\quad+\left\{-\lambda_{1,[i]}n_{3,[i]}+\mu\left\{\left(\frac{\partial\lambda_2}{\partial z}\right)_{[i]}+\left(\frac{\partial\lambda_4}{\partial x}\right)_{[i]}\right\}n_{1,[i]}+\mu\left\{\left(\frac{\partial\lambda_4}{\partial y}\right)_{[i]}+\left(\frac{\partial\lambda_3}{\partial z}\right)_{[i]}\right\}n_{2,[i]}\right.\\ &\quad\left.+2\mu\left(\frac{\partial\lambda_4}{\partial z}\right)_{[i]}n_{3,[i]}\right\}\left(\frac{\partial w}{\partial x}\right)_{[i]}\end{aligned} \tag{72}$$

This approach is also applied to the drag on the surface and the normal vector on the surface.

DOI: 10.1002/fld

6.4. Discretization of the biharmonic equation

In this subsection we describe how to solve the biharmonic equation. Equation (51) can be decomposed into two second-order partial differential equations as follows [38]:

$$-\frac{\partial^2\Theta}{\partial x^2}-\frac{\partial^2\Theta}{\partial y^2}-\frac{\partial^2\Theta}{\partial z^2}=\widetilde{\Theta} \quad \text{in } \Omega \tag{73}$$

$$-\frac{\partial^2\widetilde{\Theta}}{\partial x^2}-\frac{\partial^2\widetilde{\Theta}}{\partial y^2}-\frac{\partial^2\widetilde{\Theta}}{\partial z^2}=0 \quad \text{in } \Omega \tag{74}$$

where $\widetilde{\Theta}$ indicates the intermediate displacement. The present implementation follows a Galerkin finite element procedure. The discrete equations are given by

$$\int_\Omega \left(\frac{\partial\Theta^*}{\partial x}\frac{\partial\Theta}{\partial x}+\frac{\partial\Theta^*}{\partial y}\frac{\partial\Theta}{\partial y}+\frac{\partial\Theta^*}{\partial z}\frac{\partial\Theta}{\partial z}\right)\mathrm{d}\Omega \\ -\int_\Gamma \left(\Theta^*\frac{\partial\Theta}{\partial x}n_1+\Theta^*\frac{\partial\Theta}{\partial y}n_2+\Theta^*\frac{\partial\Theta}{\partial z}n_3\right)\mathrm{d}\Gamma=\int_\Omega \Theta^*\widetilde{\Theta}\,\mathrm{d}\Omega \tag{75}$$

$$\int_\Omega \left(\frac{\partial\widetilde{\Theta}^*}{\partial x}\frac{\partial\widetilde{\Theta}}{\partial x}+\frac{\partial\widetilde{\Theta}^*}{\partial y}\frac{\partial\widetilde{\Theta}}{\partial y}+\frac{\partial\widetilde{\Theta}^*}{\partial z}\frac{\partial\widetilde{\Theta}}{\partial z}\right)\mathrm{d}\Omega-\int_\Gamma \left(\widetilde{\Theta}^*\frac{\partial\widetilde{\Theta}}{\partial x}n_1+\widetilde{\Theta}^*\frac{\partial\widetilde{\Theta}}{\partial y}n_2+\widetilde{\Theta}^*\frac{\partial\widetilde{\Theta}}{\partial z}n_3\right)\mathrm{d}\Gamma=0 \tag{76}$$

Assuming the second terms on the left-hand side of Equations (75) and (76) to be zero, these are transformed by Equation (57) as follows:

$$\int_\Omega (b_{[a]}b_{[b]}+c_{[a]}c_{[b]}+d_{[a]}d_{[b]})\,\mathrm{d}\Omega\Theta_{[b]}\,\mathrm{d}\Gamma=\int_\Omega N_{[a]}N_{[b]}\,\mathrm{d}\Omega\widetilde{\Theta}_{[b]}, \quad a,b=1,2,3,4 \tag{77}$$

$$\int_\Omega (b_{[a]}b_{[b]}+c_{[a]}c_{[b]}+d_{[a]}d_{[b]})\,\mathrm{d}\Omega\widetilde{\Theta}_{[b]}\,\mathrm{d}\Gamma=0, \quad a,b=1,2,3,4 \tag{78}$$

Equation (78) under the boundary condition (Equations (52)–(53)) is first solved by using the HEC-MW solver to obtain the intermediate displacement $\widetilde{\Theta}$ at each x, y, z. Then, Equation (77) under the boundary condition (Equations (52)–(53)) is also solved by the HEC-MW solver to obtain the displacement Θ at each x, y, z. Using the obtained displacement values, the coordinates of all nodes are updated.

7. IMPLEMENTATION

7.1. Shape optimization algorithm

The algorithm of the shape optimization method is shown in Figure 6. Before executing the algorithm, the state equations are solved until the flow field reaches the steady state.

In the first phase of the algorithm, the state variables ($\mathbf{W}_{(k)}^{(n)}$) are calculated by using the state equations. The state equations are solved from the test of start time to the test of end time. All the nodal values of the state variables ($\mathbf{W}_{(k)}^{(n)}$) are stored at every time step.

DOI: 10.1002/fld

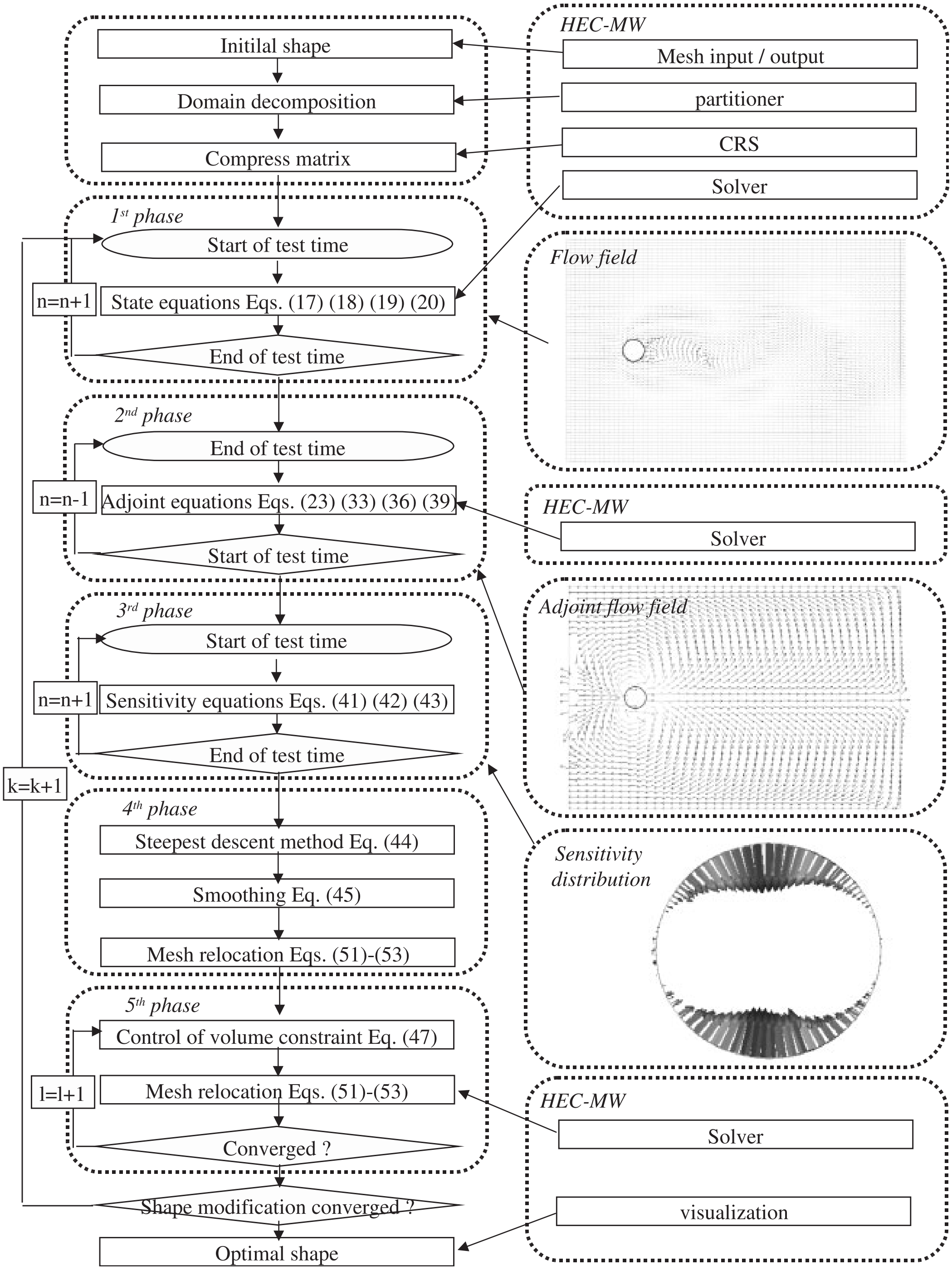


Figure 6. Optimization algorithm by the adjoint variable method.

DOI: 10.1002/fld

In the second phase of the algorithm, the adjoint variables ($\boldsymbol{\lambda}^{(n)}_{(k)}$) are calculated by Equations (23), (33), (36), and (39) from the test of start time to the test of end time. The adjoint equations, which include the time derivative, are also solved until the adjoint flow filed reaches the steady state. All the nodal values of the adjoint variables ($\boldsymbol{\lambda}^{(n)}_{(k)}$) are saved at every time step. These data are stored as files.

In the third phase, the sensitivity at every time step is calculated by using the saved files containing the adjoint and state variables. The sensitivity represents the displacement of the nodes on the surface of the object. The sensitivity must have a small value in order to robustly converge to the optimal coordinates and to avoid collapse of the mesh topology.

In the fourth phase, the shape is modified by using the time integral sensitivity. The optimization method is the steepest descent method. After that, the nodes of the mesh are relocated according to the time integral sensitivity. The node relocation is performed by using the biharmonic equation (see Section 5).

In the fifth phase, the shape is modified in order to satisfy the constraint of constant volume.

In case the shape converges to the optimum, the result is outputted. In case the shape does not converge to the optimum, the algorithm returns to the first phase.

7.2. HEC-MW

Parallel computing requires some unique techniques such as the domain decomposition, message passing, vectorization, etc.

This extra work is very burdensome for application developers; the implementation is very time consuming and often causes numerous bugs. In addition, since recently available architectures vary from PC clusters to SMP clusters, the optimization strategies to make the best use of the hardware are no longer unique. For developing the parallel finite element fluid analysis code, the present study employs 'the HEC middleware (HEC-MW)' [28], which has been developed within the project of 'Frontier Simulation Software for Industrial Science' at the Institute of Industrial Science (IIS), the University of Tokyo, as a research project of IT-program under Research Revolution 2002 organized by Ministry of Education, Culture, Sport, Science and Technology, Japan.

The HEC-MW is a hardware-independent platform, which extracts patterns of calculation processes and common interfaces of unstructured grid simulations. A program developed on a PC is automatically optimized for each high-end machine by plugging in the installed HEC-MW (Figure 7).

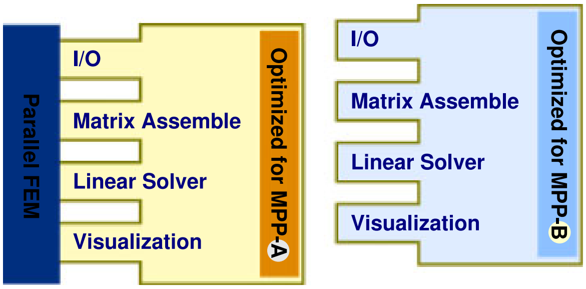


Figure 7. Utilization of HEC-MW.

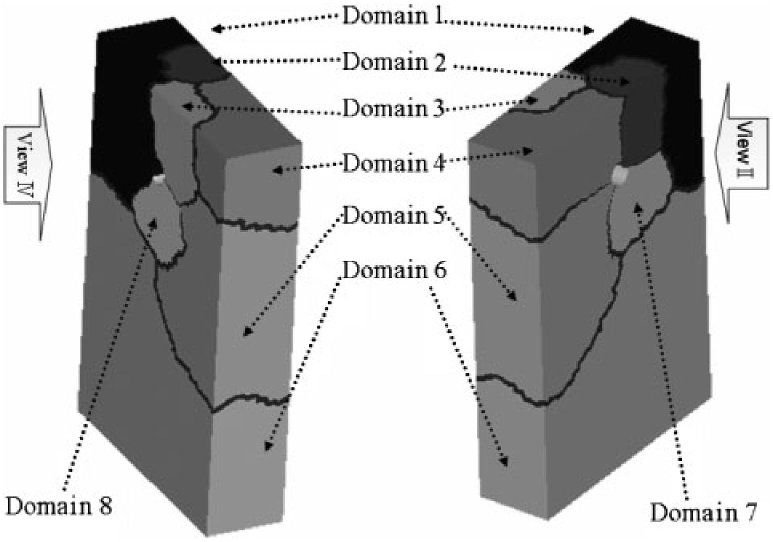


Figure 8. Domain decomposition.

By simply installing HEC-MW and calling the appropriate subroutine, complex computations can be performed. Thus, HEC-MW enables the development of portable and highly reliable simulation code.

7.3. Parallel computing by using domain decomposition and CRS format

By using HEC-MW, the calculation load is processed in parallel, on a PC cluster environment consisting of eight CPUs. As shown in Figure 8, the computational domain is divided into eight parts, which enable distributed processing on eight CPUs in the PC cluster environment.

The ‘hecmw-part’ from HEC-MW is used as a partitioner for domain decomposition. From the decomposition method, it provides METIS (based on the graph theory and RCB (based on the recursive coordinate bisection)). We selected the former.

From the point of view of memory organization, parallel computers can be either of shared memory type, where all processors share the same memory, or of distributed memory type, where each processor uses a different memory. The shared memory type organization may perform slower due to processors competing for reading or writing. Because of this fact, the distributed memory type has been adopted. Data for different domains are stored in different memories. In order to maximize performance, unnecessary data communication has to be removed. HEC-MW extracts non-zero elements from 2D matrices and compresses them to 1D vectors which do not include zero elements. This matrix storage format is called compressed row storage (CRS).

8. SHAPE OPTIMIZATION OF OBJECTS IN FLOW

8.1. Calculation model and conditions

The calculation conditions are shown in Table II. The mesh is shown in Figure 10. The mesh resolution is 121 240 nodes and 701 771 elements. The element type is 4-node tetrahedron element.

DOI: 10.1002/fld

Table II. Analysis conditions.

Parameter	Value
Density	$100.0\,\mathrm{kg/m^3}$
Viscosity	$1.0\,\mathrm{Pa\,s}$
Inlet velocity	$0.01\,\mathrm{m/s}$
Cylinder volume	$2.30\,\mathrm{m^3}$
Number of elements	701 771
Number of nodes	121 240
Time step size	0.001
Start of test time	300 s
End of test time	301 s

The cylinder surface is shown in Figure 11. Here, views I–III show viewpoints and these are associated with the ones in Figure 2. Cylinder analytical model is divided into 72 sections along circumferential direction and 120 sections along the longitudinal direction.

8.2. Verification of the robust effects on mesh deformation using Biharmonic equation and Laplace equation

To verify the robust 3D mesh deformation by using the biharmonic equation, element volumes in the mesh and the aspect ratio of element volumes by biharmonic equations are compared with those by Laplace equations. A tetrahedral element in the mesh is shown in Figure 9. Circles and a triangle in the element show the node points and the barycentric position of an element. Distances between the node points and the barycentric position are denoted as a, b, c, d, as shown in Figure 9. We assume that a is the maximum length and b is the minimum length. Aspect ratio is defined as follows:

$$\text{aspect} = \frac{b}{a} (\leqslant 1) \tag{79}$$

The element is close to regular tetrahedral in case the aspect ratio in the element is close to 1, while the element is far from the regular tetrahedral in case the aspect ratio is far from 1. The aspect ratio should be close to 1 because it preserves the calculation accuracy in fluid dynamics, and it prevents the creation of distorted unusable elements.

The problem of a moving cylinder is used as a test problem. The initial mesh is shown in Figure 10. The circular cylinder moves to the right along the x-axis (Figure 11). Figure 12 (translation, '+2') shows mesh relocation using Laplace equation. The mesh lines are partly crossed and there are triangles with negative areas. Figure 13 (translation, '+2') shows mesh relocation using biharmonic equation. Node relocations in the mesh can be executed. Comparing with node relocations using Laplace equation, the tetrahedral elements remain undistorted.

Figure 14 shows the aspect ratio distribution histogram. The horizontal axis represents the aspect ratio with respect to sections. Sections are obtained by dividing the $[0, 1]$ interval in equality-sized region of length 0.05. The vertical axis shows the member of elements for each section. Summing the elements in all sections gives the total number of elements 701 771.

By comparison with the aspect ratio histogram for the initial mesh, it was observed that the histograms for the deformed meshes are shifted towards the origin, showing an increase in the number of irregular elements. It can also be seen that the deformed histograms are very similar.

DOI: 10.1002/fld

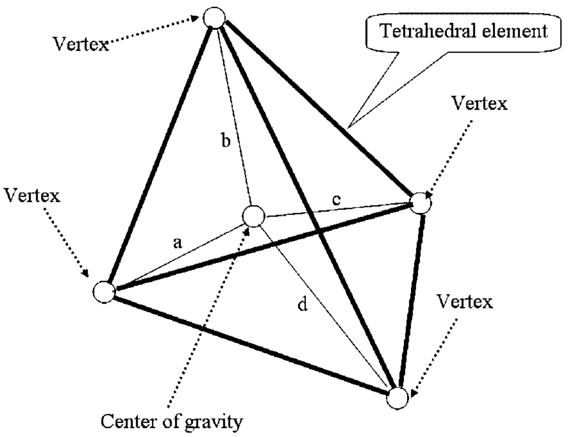


Figure 9. Tetrahedral element.

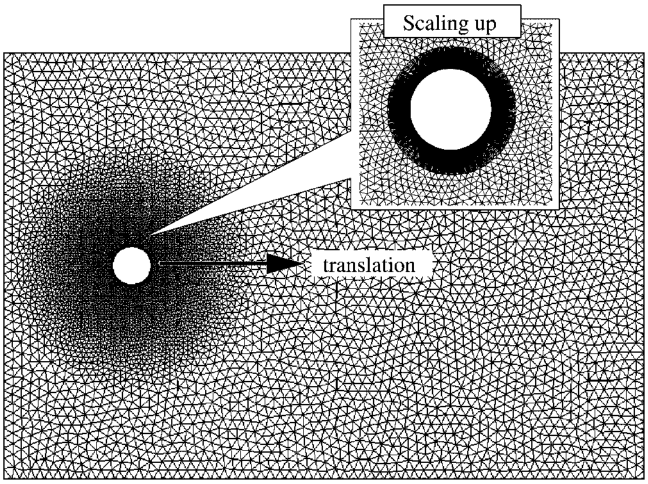


Figure 10. Mesh (initial shape).

As mesh relocations using the Laplace equation can generate elements with negative volume, resulting in the failure of the FEM, we applied the biharmonic equation, which does not exhibit such behavior.

8.3. Parallel computing using HEC-MW

PC cluster specifications are shown in Table III. In Figure 15, the graph of the calculation time is shown. The vertical axis shows the total time to reach the optimal shape. The horizontal axis shows the number of processors. In Figure 16, the vertical axis shows the speedup. The horizontal axis shows the number of processors. In the case of one CPU, 296.9 h (about 12 days) are needed.

DOI: 10.1002/fld

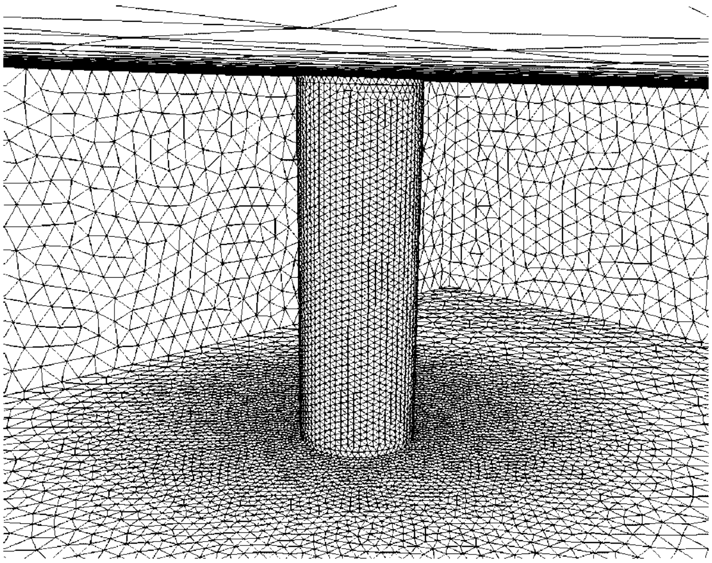


Figure 11. Mesh (initial shape) (View I).

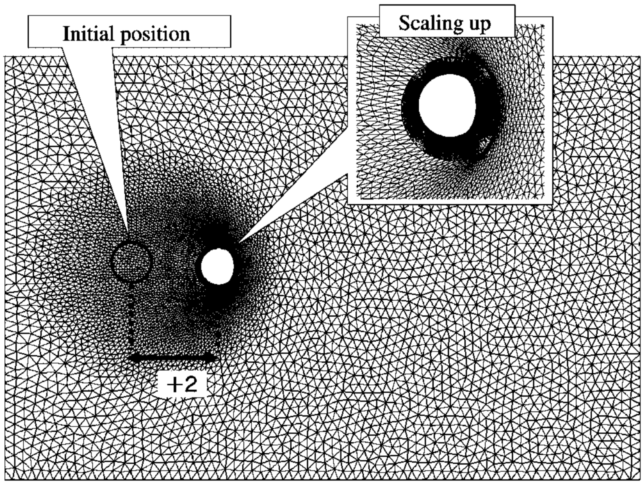


Figure 12. Mesh deformation using Laplace equations (translation +2) (View II).

In the case of eight CPUs, calculation time is 43.2 h. As shown in Figure 16, the achieved speedup is 6.6 compared with that of one CPU.

As described in the adjoint variable method algorithm, file I/O is performed at every time step. The amount of time required for solving the finite element equations (Equations (65)–(67)) by CG and for file I/O is shown in Table IV. In the eight CPU case, the file I/O time is about $\frac{1}{8}$ because the file capacity per 1 CPU reduces. The solver time is about $\frac{1}{5}$ compared with one CPU case.

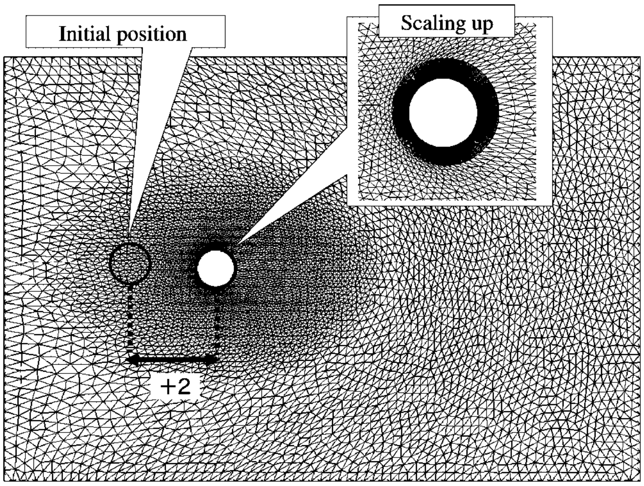


Figure 13. Mesh deformation using Biharmonic equations (translation +2) (View II).

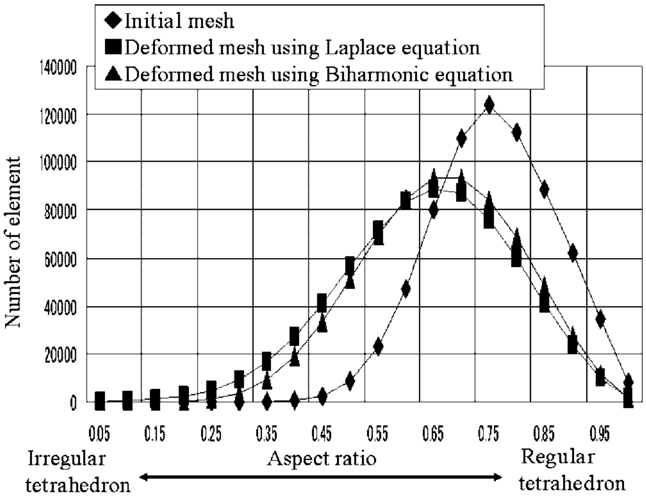


Figure 14. Aspect ratio distribution

Table III. PC cluster (for a 1PE (processor element)).

	Specs
CPU	3.0 GHz (Pentium4 630)
Memory	2 GB
HDD	80 GB
Network card	Gigabit ethernet

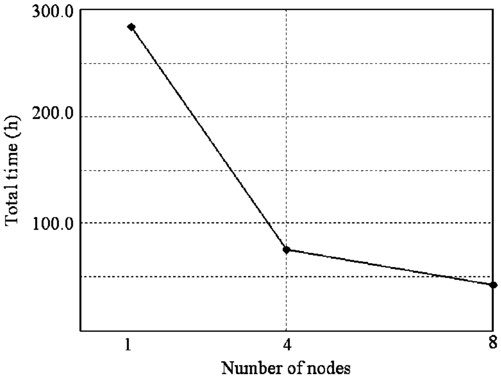


Figure 15. Total computing time (by the mesh in Figure 10).

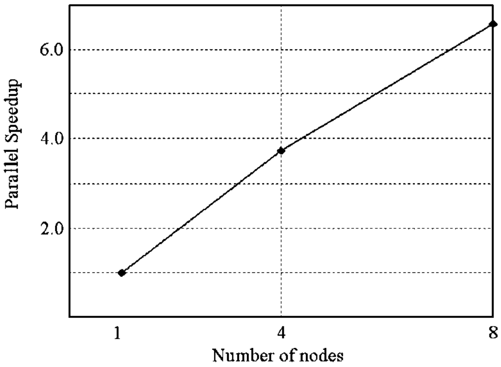


Figure 16. Parallel speedup of shape optimization algorithm (by the mesh in Figure 10).

Table IV. Processing time (per one time step) (by the mesh in Figure 9).

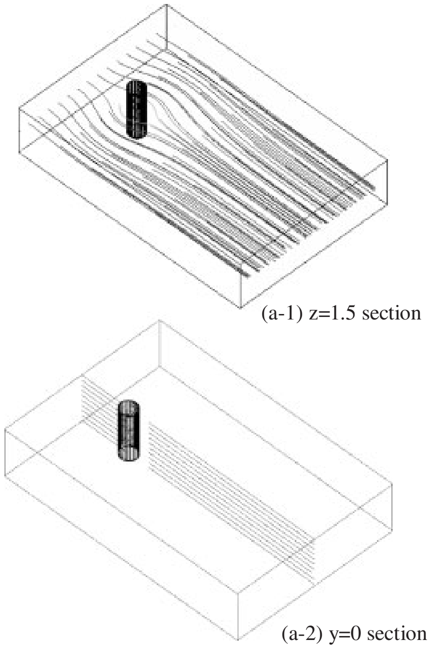
CPU	File output time	Solver time
1	0.625 s	9.20 s
8	0.084 s	2.03 s

The implementation of parallel processing is specially important to reduce the file I/O processing time in the adjoint variable method algorithm.

8.4. Calculation results

The flow on the initial shape is shown in Figure 17. The fluid velocity, which flows in the boundary Γw, decreases around the cylinder. The drag coefficient of the initial shape is calculated as follows:

$$C_{\mathrm{D}}=\frac{2\int_{\gamma}\left\{-pn_1+\mu\left(\frac{\partial u_i}{\partial x}+\frac{\partial u}{\partial x_i}\right)n_i\right\}\mathrm{d}\gamma}{\rho U^2(DL)}=\frac{2\times 0.219}{100\times(0.01)^2(1\times 3)}=14.6 \tag{80}$$



(a-1) z=1.5 section

(a-2) y=0 section

Figure 17. The streamline of the initial shape.

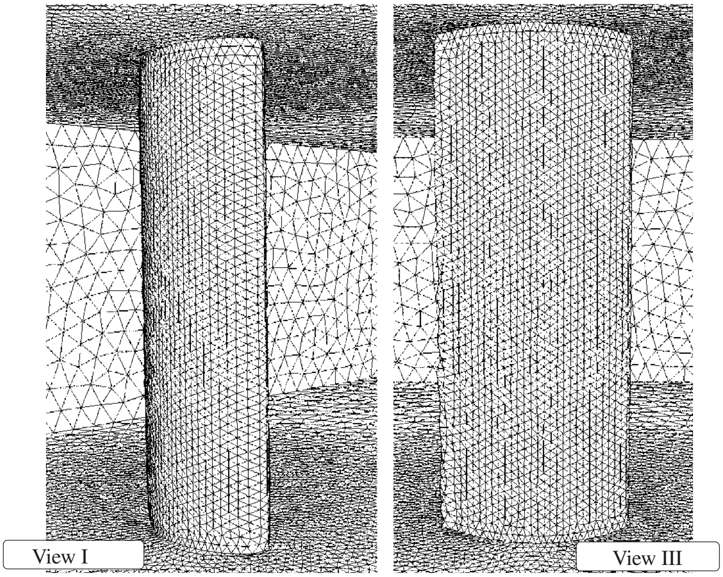
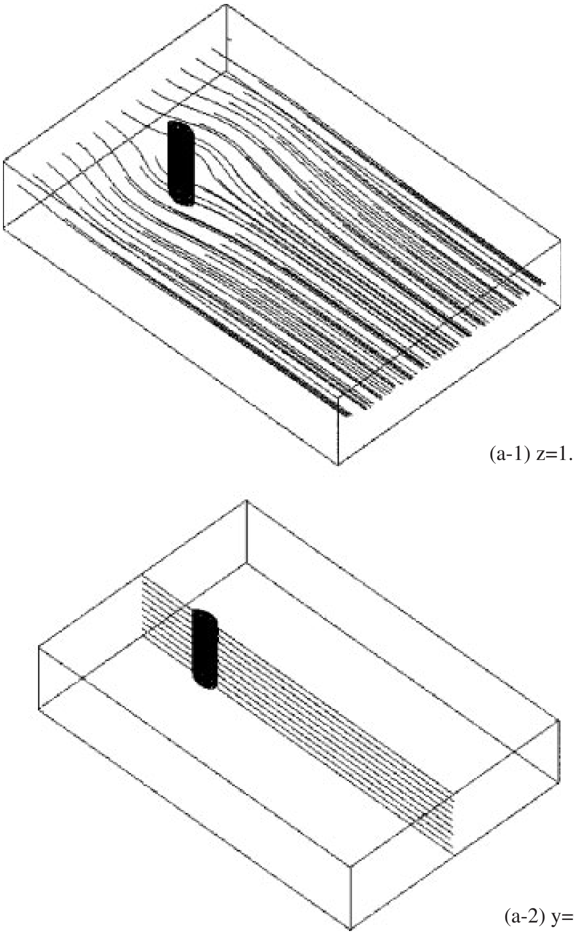


Figure 18. The mesh of the optimal shape.

DOI: 10.1002/fld



(a-1) z=1.5 section

(a-2) y=0 section

Figure 19. The flow field of the optimal shape (301.0 s).

where U is the flow velocity past the cylinder, L is the cylinder length, and D is the cylinder diameter. This coefficient is almost in agreement with the coefficient derived by theoretical equation as follows [39]:

$$C_{\mathrm{D}}=\frac{8\pi}{\frac{1}{2}-\gamma-\ln(Re/8)}=\frac{8\times 3.14}{\frac{1}{2}-0.577-\ln(1/8)}=12.54 \tag{81}$$

where γ is Euler's constant, and *Re* is the Reynolds number based on the diameter of the cylinder.

As the computed value of the drag coefficient (Equation (80)) is slightly different from the theoretical one (Equation (81)), the dependency of the error on the grid size has been investigated using three different mesh resolutions as shown in Figure 21 (nodes: 9442, elements: 46 268), in Figure 22 (nodes: 46 274, elements: 267 888) and Figure 10 (nodes: 121 240, elements: 701 771). The drag coefficients under steady flow are shown in Figure 23. The vertical axis in the graph

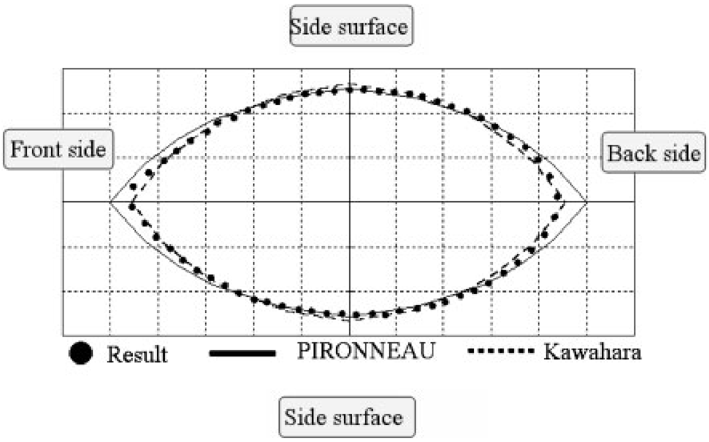


Figure 20. Comparison of the optimal shape (301.0 s).

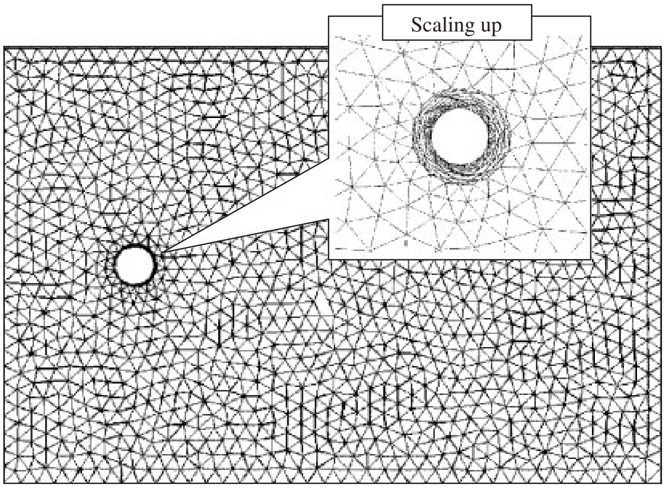


Figure 21. Mesh (node: 9442, element: 46 268).

shows the drag coefficient of the cylinder, and the horizontal axis shows the total number of nodes. The theoretical drag coefficient given by Equation (81) is shown by a thick continuous line. If the mesh resolution is low, the calculation result is not in good agreement with the theoretical result. As the grid resolution becomes higher, the calculation result gets closer to the theoretical one. Therefore, the reliability of the CFD scheme depends on the grid resolution of the finite element mesh. By using this mesh size (nodes: 121 240, elements: 701 771), we try to make an optimal shape under Stokes flow. The flow speed on the cylinder surface is zero by the non-slip condition. The flow vectors in the computational domain have symmetry along the x-axis. Uniform streamlines are generated along the z direction of the cross section ($y=0.0$). The sensitivity distribution of the initial shape is shown in Figure 24 (upper left). The optimal shape is constructed by using the

DOI: 10.1002/fld

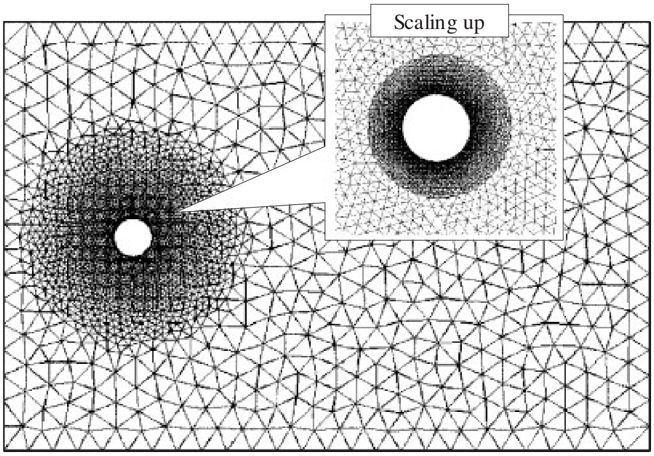


Figure 22. Mesh (node: 46 274, element: 267 888).

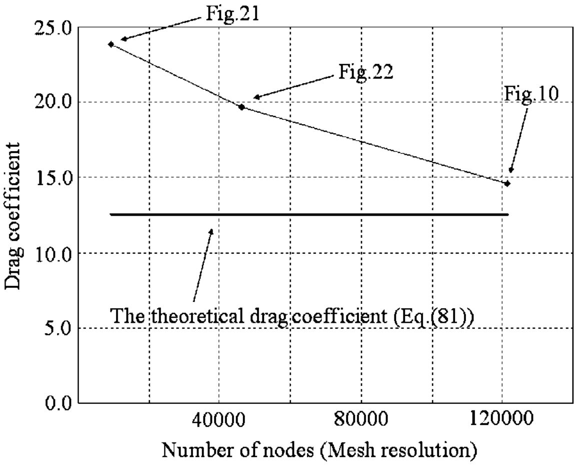


Figure 23. Drag coefficient.

sensitivity. This sensitivity is a vector on the surface from the initial shape to the optimal shape. Sensitivity distribution is elliptic and is constructed from the outside to the inside of the cylinder. Comparing with both the front and the back of the cylinder, large sensitivities are obtained at the sides of the deformed cylinder. As the shape step advances, the sensitivity distribution reaches zero and the initial shape reaches the optimal shape, as shown in Figure 24. The optimal shape is shown in Figure 17. In a cylinder shape, the viscosity considered inside the cost function is decreased whereas the pressure is increased. In a board shape, pressure is decreased whereas viscosity is increased. Therefore, an elliptical shape between the cylinder shape and the board shape is constructed to decrease both viscosity and pressure. In Figure 20, Pironneau's optimal shape is represented by the continuous line while the result obtained by this calculation is represented

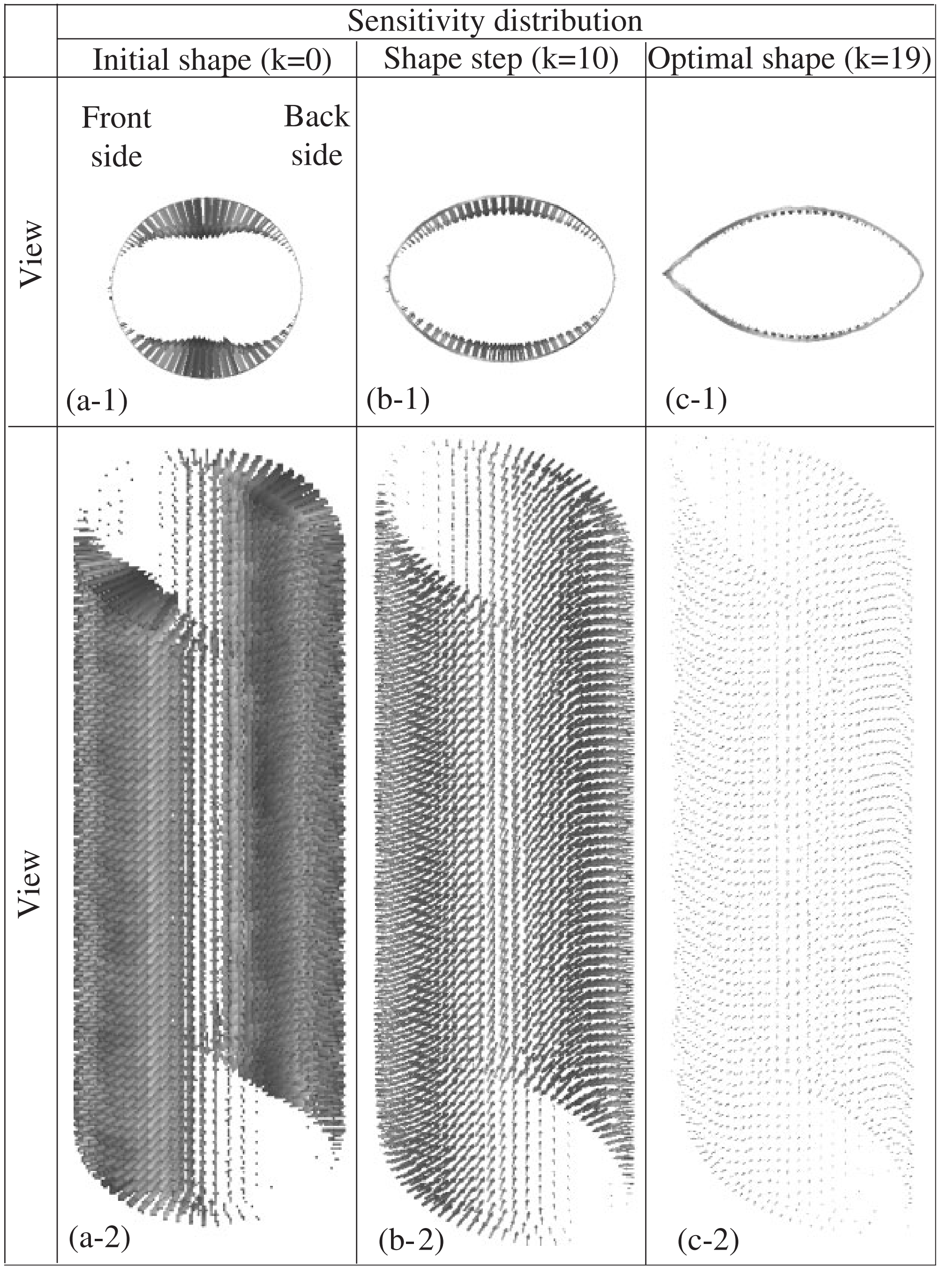


Figure 24. The history of the sensitivity distribution (by the mesh in Figure 10).

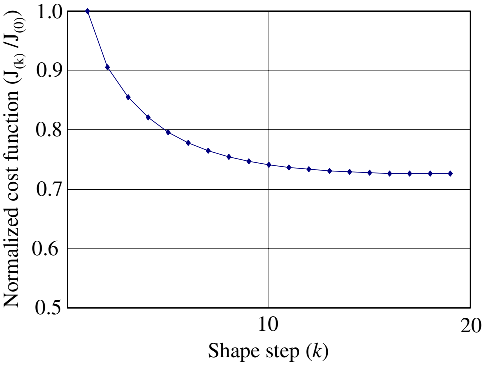


Figure 25. Convergence behavior of cost function (by the mesh in Figure 10).

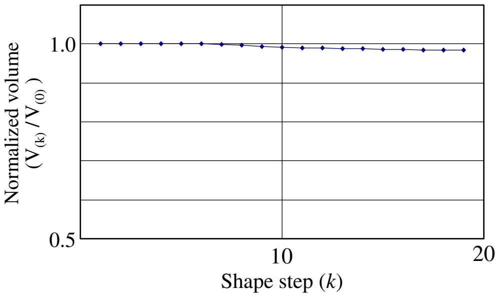


Figure 26. Volume of cylinder (by the mesh in Figure 10).

by dots. The optimal shape obtained in this study is almost in agreement with Pironneau's results under steady flow by using stokes equation, but the difference between the two lies in the sharp regions located in the front and back of Pironneau's shape (Figure 18). The streamline of the optimal shape is shown in Figure 19. The 3D flow is not generated in the case of the optimal shape. The history of the cost function and volume with respect to the shape step is shown in Figures 25 and 26. The cost function converged to the optimal value after about 20 shape steps. The cost function of the initial shape was decreased by about 25%. This reduction rate is almost in agreement with reduction rate of 23% obtained by Sano and Sakai [13].

9. CONCLUSIONS

To reduce the drag on an object under Stokes flow, a shape optimization method based on an adjoint variable method and several related techniques (parallel computing, smoothing, mesh relocation, constant volume condition) was presented. A versatile shape optimization technique was developed, which is able to robustly construct the optimal shape. Aiming at applying the proposed method to actual complex mechanical structures, the present optimization method was formulated based on the finite element method. The discretization procedure based on the FEM is more feasible for

DOI: 10.1002/fld

applying in complex structures than the one based on the finite difference methods. In general, one needs to store the pressure and the velocity information at every time step to implement the time-dependent adjoint formulation. By using the parallel computing library HEC-MW, the large data are automatically distributed to every node in the PC cluster environment. Shape optimization analysis could be executed even if the mesh was in high resolution. Biharmonic equation techniques and smoothing technique to robustly deform the mesh were also implemented in the shape optimization algorithm. A wide area search of optimal shape was enabled by the present algorithm.

Using the present shape optimization technique, a cylinder was optimized to reduce the drag. The optimal shape obtained here was almost in agreement with the one obtained by Pironneau. By the shape optimization, the drag on the cylinder was decreased by about 25%. This reduction ratio was in reasonable agreement with that presented in the literature.

APPENDIX A: TRANSFORMATION OF LAGRANGE FUNCTION

The Lagrange function (Equation (14)) is transformed to derive the adjoint equation and the sensitivity equation. By applying the Gauss green theorem, the second term of Equation (13) becomes as follows:

$$\begin{aligned}
&\frac{1}{\rho}\int_{t_\mathrm{s}}^{t_\mathrm{e}}\int_{\Omega(\mathbf{S})}\lambda_1\left(\frac{\partial u}{\partial x}+\frac{\partial v}{\partial y}+\frac{\partial w}{\partial z}\right)\mathrm{d}\Omega\,\mathrm{d}t\\
&\quad=\frac{1}{\rho}\int_{t_\mathrm{s}}^{t_\mathrm{e}}\int_{\Gamma(\mathbf{S})}\lambda_1 u_i n_i\,\mathrm{d}\Gamma\,\mathrm{d}t-\frac{1}{\rho}\int_{t_\mathrm{s}}^{t_\mathrm{e}}\int_{\Omega(\mathbf{S})}\frac{\partial\lambda_1}{\partial x_i}u_i\,\mathrm{d}\Omega\,\mathrm{d}t,\quad i=1,2,3
\end{aligned}\tag{A1}$$

The other term of Equation (14) is as follows:

$$\begin{aligned}
&\int_{t_\mathrm{s}}^{t_\mathrm{e}}\int_{\Omega(\mathbf{S})}\lambda_{i+1}\left\{-\frac{\partial u_i}{\partial t}-\frac{1}{\rho}\frac{\partial p}{\partial x_i}+v\frac{\partial}{\partial x_j}\left(\frac{\partial u_j}{\partial x_i}+\frac{\partial u_i}{\partial x_j}\right)\right\}\mathrm{d}\Omega\,\mathrm{d}t\\
&=-\int_{t_\mathrm{s}}^{t_\mathrm{e}}\int_{\Omega(\mathbf{S})}\lambda_{i+1}\frac{\partial u_i}{\partial t}\,\mathrm{d}\Omega\,\mathrm{d}t-\int_{t_\mathrm{s}}^{t_\mathrm{e}}\int_{\Gamma(\mathbf{S})}\lambda_{i+1}pn_i\frac{1}{\rho}\,\mathrm{d}\Gamma\,\mathrm{d}t+\int_{t_\mathrm{s}}^{t_\mathrm{e}}\int_{\Omega(\mathbf{S})}\left(\frac{\partial\lambda_2}{\partial x}+\frac{\partial\lambda_3}{\partial y}+\frac{\partial\lambda_4}{\partial z}\right)p\frac{1}{\rho}\,\mathrm{d}\Omega\,\mathrm{d}t\\
&\quad+\int_{t_\mathrm{s}}^{t_\mathrm{e}}\int_{\Gamma(\mathbf{S})}v\lambda_{i+1}\left(\frac{\partial u_j}{\partial x_i}+\frac{\partial u_i}{\partial x_j}\right)n_j\,\mathrm{d}\Gamma\,\mathrm{d}t-v\int_{t_\mathrm{s}}^{t_\mathrm{e}}\int_{\Omega(\mathbf{S})}\frac{\partial\lambda_{i+1}}{\partial x_j}\left(\frac{\partial u_j}{\partial x_i}+\frac{\partial u_i}{\partial x_j}\right)\mathrm{d}\Omega\,\mathrm{d}t\\
&=-\int_{\Omega(\mathbf{S})}[\lambda_{i+1}u_i]_{t_\mathrm{s}}^{t_\mathrm{e}}\,\mathrm{d}\Omega+\int_{t_\mathrm{s}}^{t_\mathrm{e}}\int_{\Omega(\mathbf{S})}\frac{\partial\lambda_{i+1}}{\partial t}u_i\,\mathrm{d}\Omega\,\mathrm{d}t+\int_{t_\mathrm{s}}^{t_\mathrm{e}}\int_{\Omega(\mathbf{S})}\left(\frac{\partial\lambda_2}{\partial x}+\frac{\partial\lambda_3}{\partial y}+\frac{\partial\lambda_4}{\partial z}\right)p\frac{1}{\rho}\,\mathrm{d}\Omega\,\mathrm{d}t\\
&\quad+\int_{t_\mathrm{s}}^{t_\mathrm{e}}\int_{\Gamma(\mathbf{S})}\lambda_{i+1}\left\{-\frac{p}{\rho}n_i+v\left(\frac{\partial u_j}{\partial x_i}+\frac{\partial u_i}{\partial x_j}\right)n_j\right\}\mathrm{d}\Gamma\,\mathrm{d}t-v\int_{t_\mathrm{s}}^{t_\mathrm{e}}\int_{\Gamma(\mathbf{S})}\frac{\partial\lambda_{i+1}}{\partial x_j}(u_jn_i+u_in_j)\,\mathrm{d}\Gamma\,\mathrm{d}t\\
&\quad+v\int_{t_\mathrm{s}}^{t_\mathrm{e}}\int_{\Omega(\mathbf{S})}\left(\frac{\partial}{\partial x_j}\frac{\partial\lambda_{i+1}}{\partial x_j}u_i+\frac{\partial}{\partial x_i}\frac{\partial\lambda_{i+1}}{\partial x_j}u_j\right)\mathrm{d}\Omega\,\mathrm{d}t,\quad i,j=1,2,3
\end{aligned}\tag{A2}$$

DOI: 10.1002/fld

The p, u, v, and w terms in the integrand of Ω term are arranged by using the following equation:

$$\begin{aligned}
&\nu\int_{t_\mathrm{s}}^{t_\mathrm{e}}\int_{\Gamma(\mathbf{S})}\frac{\partial\lambda_{i+1}}{\partial x_j}(u_i n_j+u_j n_i)\,\mathrm{d}\Gamma\,\mathrm{d}t\\
&\quad=\nu\int_{t_\mathrm{s}}^{t_\mathrm{e}}\int_{\Gamma(\mathbf{S})}\left(\frac{\partial\lambda_{i+1}}{\partial x_j}u_i n_j+\frac{\partial\lambda_{j+1}}{\partial x_i}u_i n_j\right)\mathrm{d}\Gamma\,\mathrm{d}t\\
&\quad=\nu\int_{t_\mathrm{s}}^{t_\mathrm{e}}\int_{\Gamma(\mathbf{S})}\left(\frac{\partial\lambda_{i+1}}{\partial x_j}+\frac{\partial\lambda_{j+1}}{\partial x_i}\right)u_i n_j\,\mathrm{d}\Gamma\,\mathrm{d}t
\end{aligned}\tag{A3}$$

$$\begin{aligned}
&\nu\int_{t_\mathrm{s}}^{t_\mathrm{e}}\int_{\Omega(\mathbf{S})}\left(\frac{\partial}{\partial x_j}\frac{\partial\lambda_{i+1}}{\partial x_j}u_i+\frac{\partial}{\partial x_i}\frac{\partial\lambda_{i+1}}{\partial x_j}u_j\right)\mathrm{d}\Omega\,\mathrm{d}t\\
&\quad=\nu\int_{t_\mathrm{s}}^{t_\mathrm{e}}\int_{\Omega(\mathbf{S})}\left(\frac{\partial}{\partial x_j}\frac{\partial\lambda_{i+1}}{\partial x_j}u_i+\frac{\partial}{\partial x_j}\frac{\partial\lambda_{j+1}}{\partial x_i}u_i\right)\mathrm{d}\Omega\,\mathrm{d}t\\
&\quad=\nu\int_{t_\mathrm{s}}^{t_\mathrm{e}}\int_{\Omega(\mathbf{S})}u_i\frac{\partial}{\partial x_j}\left(\frac{\partial\lambda_{i+1}}{\partial x_j}+\frac{\partial\lambda_{j+1}}{\partial x_i}\right)\mathrm{d}\Omega\,\mathrm{d}t
\end{aligned}\tag{A4}$$

Equation (15) is derived after the transformation and the arrangement.

APPENDIX B: TRANSFORMATION OF SENSITIVITY EQUATION

The Lagrange function (Equation (40)) is differentiated with respect to space coordinates in order to derive the sensitivity equation:

$$\begin{aligned}
\frac{\partial L(\mathbf{S})}{\partial x_k}=&-\int_{t_\mathrm{s}}^{t_\mathrm{e}}\int_{\gamma(\mathbf{S})}\left\{-\frac{1}{\rho}\frac{\partial p}{\partial x_k}n_1+\nu\frac{\partial}{\partial x_k}\left(\frac{\partial u}{\partial x_j}+\frac{\partial u_j}{\partial x}\right)n_j\right\}\mathrm{d}\gamma\,\mathrm{d}t\\
&-\int_{t_\mathrm{s}}^{t_\mathrm{e}}\int_{\gamma(\mathbf{S})}\left\{-\frac{p}{\rho}\frac{\partial n_1}{\partial x_k}+\nu\left(\frac{\partial u}{\partial x_j}+\frac{\partial u_j}{\partial x}\right)\frac{\partial n_j}{\partial x_k}\right\}\mathrm{d}\gamma\,\mathrm{d}t\\
&+\int_{t_\mathrm{s}}^{t_\mathrm{e}}\int_{\Gamma(\mathbf{S})}\frac{\partial\lambda_{i+1}}{\partial x_k}\left\{-\frac{p}{\rho}n_i+\nu\left(\frac{\partial u_i}{\partial x_j}+\frac{\partial u_j}{\partial x_i}\right)n_j\right\}\mathrm{d}\Gamma\,\mathrm{d}t\\
&+\int_{t_\mathrm{s}}^{t_\mathrm{e}}\int_{\Gamma(\mathbf{S})}\lambda_{i+1}\left\{-\frac{1}{\rho}\frac{\partial p}{\partial x_k}n_i+\nu\frac{\partial}{\partial x_k}\left(\frac{\partial u_i}{\partial x_j}+\frac{\partial u_j}{\partial x_i}\right)n_j\right\}\mathrm{d}\Gamma\,\mathrm{d}t\\
&+\int_{t_\mathrm{s}}^{t_\mathrm{e}}\int_{\Gamma(\mathbf{S})}\lambda_{i+1}\left\{-p\frac{1}{\rho}\frac{\partial n_i}{\partial x_k}+\nu\left(\frac{\partial u_i}{\partial x_j}+\frac{\partial u_j}{\partial x_i}\right)\frac{\partial n_j}{\partial x_k}\right\}\mathrm{d}\Gamma\,\mathrm{d}t\\
&-\int_{t_\mathrm{s}}^{t_\mathrm{e}}\int_{\Gamma(\mathbf{S})}\frac{\partial u_i}{\partial x_k}\left\{-\frac{\lambda_1}{\rho}n_i+\nu\left(\frac{\partial\lambda_{i+1}}{\partial x_j}+\frac{\partial\lambda_{j+1}}{\partial x_i}\right)n_j\right\}\mathrm{d}\Gamma\,\mathrm{d}t
\end{aligned}$$

$$
\begin{aligned}
&-\int_{t_\mathrm{s}}^{t_\mathrm{e}}\int_{\Gamma(\mathbf{S})} u_i\left\{-\frac{\partial\lambda_1}{\partial x_k}n_i+v\frac{\partial}{\partial x_k}\left(\frac{\partial\lambda_{i+1}}{\partial x_j}+\frac{\partial\lambda_{j+1}}{\partial x_i}\right)n_j\right\}\mathrm{d}\Gamma\,\mathrm{d}t\\
&-\int_{t_\mathrm{s}}^{t_\mathrm{e}}\int_{\Gamma(\mathbf{S})} u_i\left\{-\frac{\lambda_1}{\rho}\frac{\partial n_i}{\partial x_k}+v\left(\frac{\partial\lambda_{i+1}}{\partial x_j}+\frac{\partial\lambda_{j+1}}{\partial x_i}\right)\frac{\partial n_j}{\partial x_k}\right\}\mathrm{d}\Gamma\,\mathrm{d}t\\
&+\int_{t_\mathrm{s}}^{t_\mathrm{e}}\int_{\Omega(\mathbf{S})}\frac{\partial p}{\partial x_k}\left(\frac{\partial\lambda_2}{\partial x}+\frac{\partial\lambda_3}{\partial y}+\frac{\partial\lambda_4}{\partial z}\right)\mathrm{d}\Omega\,\mathrm{d}t+\int_{t_\mathrm{s}}^{t_\mathrm{e}}\int_{\Omega(\mathbf{S})}\frac{p}{\rho}\frac{\partial}{\partial x_k}\left(\frac{\partial\lambda_2}{\partial x}+\frac{\partial\lambda_3}{\partial y}+\frac{\partial\lambda_4}{\partial z}\right)\mathrm{d}\Omega\,\mathrm{d}t\\
&+\int_{t_\mathrm{s}}^{t_\mathrm{e}}\int_{\Omega(\mathbf{S})}\frac{\partial u_i}{\partial x_k}\left\{\frac{\partial\lambda_{i+1}}{\partial t}-\frac{1}{\rho}\frac{\partial\lambda_1}{\partial x_i}+v\frac{\partial}{\partial x_j}\left(\frac{\partial\lambda_{j+1}}{\partial x_i}+\frac{\partial\lambda_{i+1}}{\partial x_j}\right)\right\}\mathrm{d}\Omega\,\mathrm{d}t\\
&-\int_{\Omega(\mathbf{S})}\left[\frac{\partial\lambda_{i+1}}{\partial x_k}u_i+\lambda_{i+1}\frac{\partial u_i}{\partial x_k}\right]_{t_\mathrm{s}}^{t_\mathrm{e}}\mathrm{d}\Omega\\
&+\int_{t_\mathrm{s}}^{t_\mathrm{e}}\int_{\Omega(\mathbf{S})}u_i\frac{\partial}{\partial x_k}\left\{\frac{\partial\lambda_{i+1}}{\partial t}-\frac{1}{\rho}\frac{\partial\lambda_1}{\partial x_i}+v\frac{\partial}{\partial x_j}\left(\frac{\partial\lambda_{j+1}}{\partial x_i}+\frac{\partial\lambda_{i+1}}{\partial x_j}\right)\right\}\mathrm{d}\Omega\,\mathrm{d}t=0\\
&k,i,j=1,2,3
\end{aligned}
\tag{B1}
$$

The 10th term and the 12th term in the above equation can be transformed as follows:

$$
\begin{aligned}
&\int_{t_\mathrm{s}}^{t_\mathrm{e}}\int_{\Omega(\mathbf{S})}\frac{p}{\rho}\frac{\partial}{\partial x_k}\left(\frac{\partial\lambda_2}{\partial x}+\frac{\partial\lambda_3}{\partial y}+\frac{\partial\lambda_4}{\partial z}\right)\mathrm{d}\Omega\,\mathrm{d}t\\
&\quad+\int_{t_\mathrm{s}}^{t_\mathrm{e}}\int_{\Omega(\mathbf{S})}u_i\frac{\partial}{\partial x_k}\left\{\frac{\partial\lambda_{i+1}}{\partial t}-\frac{1}{\rho}\frac{\partial\lambda_1}{\partial x_i}+v\frac{\partial}{\partial x_j}\left(\frac{\partial\lambda_{j+1}}{\partial x_i}+\frac{\partial\lambda_{i+1}}{\partial x_j}\right)\right\}\mathrm{d}\Omega\,\mathrm{d}t\\
&=\int_{t_\mathrm{s}}^{t_\mathrm{e}}\int_{\Omega(\mathbf{S})}\frac{p}{\rho}\left(\frac{\partial}{\partial x}\frac{\partial\lambda_2}{\partial x_k}+\frac{\partial}{\partial y}\frac{\partial\lambda_3}{\partial x_k}+\frac{\partial}{\partial z}\frac{\partial\lambda_4}{\partial x_k}\right)\mathrm{d}\Omega\,\mathrm{d}t+\int_{t_\mathrm{s}}^{t_\mathrm{e}}\int_{\Omega(\mathbf{S})}u_i\frac{\partial}{\partial x_k}\frac{\partial\lambda_{i+1}}{\partial t}\mathrm{d}\Omega\,\mathrm{d}t\\
&\quad-\int_{t_\mathrm{s}}^{t_\mathrm{e}}\int_{\Omega(\mathbf{S})}u_i\frac{\partial}{\partial x_i}\frac{\partial\lambda_1}{\partial x_k}\frac{1}{\rho}\mathrm{d}\Omega\,\mathrm{d}t+v\int_{t_\mathrm{s}}^{t_\mathrm{e}}\int_{\Omega(\mathbf{S})}u_i\left\{\frac{\partial}{\partial x_j}\left(\frac{\partial}{\partial x_i}\frac{\partial\lambda_{j+1}}{\partial x_k}+\frac{\partial}{\partial x_j}\frac{\partial\lambda_{i+1}}{\partial x_k}\right)\right\}\mathrm{d}\Omega\,\mathrm{d}t\\
&=\int_{t_\mathrm{s}}^{t_\mathrm{e}}\int_{\Gamma(\mathbf{S})}\frac{p}{\rho}\frac{\partial\lambda_{i+1}}{\partial x_k}n_i\,\mathrm{d}\Gamma\,\mathrm{d}t-\int_{t_\mathrm{s}}^{t_\mathrm{e}}\int_{\Omega(\mathbf{S})}\frac{1}{\rho}\frac{\partial\lambda_{i+1}}{\partial x_k}\frac{\partial p}{\partial x_i}\mathrm{d}\Omega\,\mathrm{d}t+\int_{t_\mathrm{s}}^{t_\mathrm{e}}\int_{\Omega(\mathbf{S})}u_i\frac{\partial}{\partial t}\frac{\partial\lambda_{i+1}}{\partial x_k}\mathrm{d}\Omega\,\mathrm{d}t\\
&\quad-\int_{t_\mathrm{s}}^{t_\mathrm{e}}\int_{\Gamma(\mathbf{S})}u_i\frac{\partial\lambda_1}{\partial x_k}\frac{1}{\rho}n_i\,\mathrm{d}\Gamma\,\mathrm{d}t+\int_{t_\mathrm{s}}^{t_\mathrm{e}}\int_{\Omega(\mathbf{S})}\left(\frac{\partial u}{\partial x}+\frac{\partial v}{\partial y}+\frac{\partial w}{\partial z}\right)\frac{\partial\lambda_1}{\partial x_k}\frac{1}{\rho}\mathrm{d}\Omega\,\mathrm{d}t\\
&\quad+v\int_{t_\mathrm{s}}^{t_\mathrm{e}}\int_{\Gamma(\mathbf{S})}u_i\left(\frac{\partial}{\partial x_i}\frac{\partial\lambda_{j+1}}{\partial x_k}+\frac{\partial}{\partial x_j}\frac{\partial\lambda_{i+1}}{\partial x_k}\right)n_j\,\mathrm{d}\Gamma\,\mathrm{d}t\\
&\quad-v\int_{t_\mathrm{s}}^{t_\mathrm{e}}\int_{\Omega(\mathbf{S})}\frac{\partial u_i}{\partial x_j}\left(\frac{\partial}{\partial x_i}\frac{\partial\lambda_{j+1}}{\partial x_k}+\frac{\partial}{\partial x_j}\frac{\partial\lambda_{i+1}}{\partial x_k}\right)\mathrm{d}\Omega\,\mathrm{d}t
\end{aligned}
$$

$$
\begin{aligned}
&= \int_{t_\mathrm{s}}^{t_\mathrm{e}} \int_{\Gamma(\mathbf{S})} \frac{p}{\rho} \frac{\partial \lambda_{i+1}}{\partial x_k} n_i \, \mathrm{d}\Gamma \, \mathrm{d}t - \int_{t_\mathrm{s}}^{t_\mathrm{e}} \int_{\Omega(\mathbf{S})} \frac{\partial \lambda_{i+1}}{\partial x_k} \frac{\partial p}{\partial x_i} \frac{1}{\rho} \mathrm{d}\Omega \, \mathrm{d}t + \int_{\Omega(\mathbf{S})} \left[u_i \frac{\partial \lambda_{i+1}}{\partial x_k} \right]_{t_\mathrm{s}}^{t_\mathrm{e}} \mathrm{d}\Omega \\
&\quad - \int_{t_\mathrm{s}}^{t_\mathrm{e}} \int_{\Omega(\mathbf{S})} \frac{\partial u_i}{\partial t} \frac{\partial \lambda_{i+1}}{\partial x_k} \mathrm{d}\Omega \, \mathrm{d}t - \int_{t_\mathrm{s}}^{t_\mathrm{e}} \int_{\Omega(\mathbf{S})} u_i \frac{\partial}{\partial x_i} \frac{\partial \lambda_1}{\partial x_k} \frac{1}{\rho} \mathrm{d}\Omega \, \mathrm{d}t \\
&\quad + \int_{t_\mathrm{s}}^{t_\mathrm{e}} \int_{\Omega(\mathbf{S})} \left(\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} + \frac{\partial w}{\partial z} \right) \frac{\partial \lambda_1}{\partial x_k} \frac{1}{\rho} \mathrm{d}\Omega \, \mathrm{d}t \\
&\quad + v \int_{t_\mathrm{s}}^{t_\mathrm{e}} \int_{\Gamma(\mathbf{S})} u_i \left(\frac{\partial}{\partial x_i} \frac{\partial \lambda_{j+1}}{\partial x_k} + \frac{\partial}{\partial x_j} \frac{\partial \lambda_{i+1}}{\partial x_k} \right) n_j \, \mathrm{d}\Gamma \, \mathrm{d}t \\
&\quad - v \int_{t_\mathrm{s}}^{t_\mathrm{e}} \int_{\Gamma(\mathbf{S})} \frac{\partial u_i}{\partial x_j} \left(\frac{\partial \lambda_{j+1}}{\partial x_k} n_i + \frac{\partial \lambda_{i+1}}{\partial x_k} n_j \right) \mathrm{d}\Gamma \, \mathrm{d}t \\
&\quad + v \int_{t_\mathrm{s}}^{t_\mathrm{e}} \int_{\Omega(\mathbf{S})} \left(\frac{\partial}{\partial x_i} \frac{\partial u_i}{\partial x_j} \frac{\partial \lambda_{i+1}}{\partial x_k} + \frac{\partial}{\partial x_j} \frac{\partial u_i}{\partial x_j} \frac{\partial \lambda_{i+1}}{\partial x_k} \right) \mathrm{d}\Omega \, \mathrm{d}t \\
&= \int_{t_\mathrm{s}}^{t_\mathrm{e}} \int_{\Gamma(\mathbf{S})} \left\{ u_i \left(-\frac{1}{\rho} \frac{\partial \lambda_1}{\partial x_k} n_i + v \frac{\partial}{\partial x_k} \left(\frac{\partial \lambda_{j+1}}{\partial x_i} + \frac{\partial \lambda_{i+1}}{\partial x_j} \right) n_j \right) \right\} \mathrm{d}\Gamma \, \mathrm{d}t \\
&\quad - \int_{t_\mathrm{s}}^{t_\mathrm{e}} \int_{\Gamma(\mathbf{S})} \frac{\partial \lambda_{i+1}}{\partial x_k} \left\{ -\frac{p}{\rho} n_i + v \left(\frac{\partial u_j}{\partial x_i} + \frac{\partial u_i}{\partial x_j} \right) n_j \right\} \mathrm{d}\Gamma \, \mathrm{d}t \\
&\quad + \int_{t_\mathrm{s}}^{t_\mathrm{e}} \int_{\Omega(\mathbf{S})} \left(\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} + \frac{\partial w}{\partial z} \right) \frac{\partial \lambda_1}{\partial x_k} \frac{1}{\rho} \mathrm{d}\Omega \, \mathrm{d}t \\
&\quad + \int_{t_\mathrm{s}}^{t_\mathrm{e}} \int_{\Omega(\mathbf{S})} \frac{\partial \lambda_{i+1}}{\partial x_k} \left\{ -\frac{\partial u_i}{\partial t} - \frac{1}{\rho} \frac{\partial p}{\partial x_i} + v \frac{\partial}{\partial x_j} \left(\frac{\partial u_j}{\partial x_i} + \frac{\partial u_i}{\partial x_j} \right) \right\} \mathrm{d}\Omega \, \mathrm{d}t \\
&\quad + \int_{\Omega(\mathbf{S})} \left[u_i \frac{\partial \lambda_{i+1}}{\partial x_k} \right]_{t_\mathrm{s}}^{t_\mathrm{e}} \mathrm{d}\Omega, \quad k, j, j = 1, 2, 3
\end{aligned}
\tag{B2}
$$

By using Equation (B2), Equation (B1) is transformed as follows:

$$
\begin{aligned}
\frac{\partial L(\mathbf{S})}{\partial x_k} &= - \int_{t_\mathrm{s}}^{t_\mathrm{e}} \int_{\gamma(\mathbf{S})} \left\{ -\frac{1}{\rho} \frac{\partial p}{\partial x_k} n_1 + v \frac{\partial}{\partial x_k} \left(\frac{\partial u}{\partial x_j} + \frac{\partial u_j}{\partial x} \right) n_j \right\} \mathrm{d}\gamma \, \mathrm{d}t \\
&\quad - \int_{t_\mathrm{s}}^{t_\mathrm{e}} \int_{\gamma(\mathbf{S})} \left\{ -\frac{p}{\rho} \frac{\partial n_1}{\partial x_k} + v \left(\frac{\partial u}{\partial x_j} + \frac{\partial u_j}{\partial x} \right) \frac{\partial n_j}{\partial x_k} \right\} \mathrm{d}\gamma \, \mathrm{d}t \\
&\quad + \int_{t_\mathrm{s}}^{t_\mathrm{e}} \int_{\Gamma(\mathbf{S})} \lambda_{i+1} \left\{ -\frac{1}{\rho} \frac{\partial p}{\partial x_k} n_i + v \frac{\partial}{\partial x_k} \left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} \right) n_j \right\} \mathrm{d}\Gamma \, \mathrm{d}t \\
&\quad + \int_{t_\mathrm{s}}^{t_\mathrm{e}} \int_{\Gamma(\mathbf{S})} \lambda_{i+1} \left\{ -\frac{p}{\rho} \frac{\partial n_i}{\partial x_k} + v \left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} \right) \frac{\partial n_j}{\partial x_k} \right\} \mathrm{d}\Gamma \, \mathrm{d}t
\end{aligned}
$$

$$
\begin{aligned}
&-\int_{t_\mathrm{s}}^{t_\mathrm{e}}\int_{\Gamma(\mathbf{S})}\frac{\partial u_i}{\partial x_k}\left\{-\frac{\lambda_1}{\rho}n_i+\nu\left(\frac{\partial\lambda_{i+1}}{\partial x_j}+\frac{\partial\lambda_{j+1}}{\partial x_i}\right)n_j\right\}\mathrm{d}\Gamma\,\mathrm{d}t\\
&-\int_{t_\mathrm{s}}^{t_\mathrm{e}}\int_{\Gamma(\mathbf{S})}u_i\left\{-\frac{\lambda_1}{\rho}\frac{\partial n_i}{\partial x_k}+\nu\left(\frac{\partial\lambda_{i+1}}{\partial x_j}+\frac{\partial\lambda_{j+1}}{\partial x_i}\right)\frac{\partial n_j}{\partial x_k}\right\}\mathrm{d}\Gamma\,\mathrm{d}t\\
&-\int_{t_\mathrm{s}}^{t_\mathrm{e}}\int_{\Omega(\mathbf{S})}\frac{\partial p}{\partial x_k}\left(\frac{\partial\lambda_2}{\partial x}+\frac{\partial\lambda_3}{\partial y}+\frac{\partial\lambda_4}{\partial z}\right)\frac{1}{\rho}\mathrm{d}\Omega\,\mathrm{d}t+\int_{t_\mathrm{s}}^{t_\mathrm{e}}\int_{\Omega(\mathbf{S})}\left(\frac{\partial u}{\partial x}+\frac{\partial v}{\partial y}+\frac{\partial w}{\partial z}\right)\frac{\partial\lambda_1}{\partial x_k}\frac{1}{\rho}\mathrm{d}\Omega\,\mathrm{d}t\\
&+\int_{t_\mathrm{s}}^{t_\mathrm{e}}\int_{\Omega(\mathbf{S})}\frac{\partial u_i}{\partial x_k}\left\{\frac{\partial\lambda_{i+1}}{\partial t}-\frac{1}{\rho}\frac{\partial\lambda_1}{\partial x_i}+\nu\frac{\partial}{\partial x_j}\left(\frac{\partial\lambda_{j+1}}{\partial x_i}+\frac{\partial\lambda_{i+1}}{\partial x_j}\right)\right\}\mathrm{d}\Omega\,\mathrm{d}t-\int_{\Omega(\mathbf{S})}\left[\lambda_{i+1}\frac{\partial u_i}{\partial x_k}\right]_{t_\mathrm{s}}^{t_\mathrm{e}}\mathrm{d}\Omega\\
&+\int_{t_\mathrm{s}}^{t_\mathrm{e}}\int_{\Omega(\mathbf{S})}\frac{\partial\lambda_{i+1}}{\partial x_k}\left\{-\frac{\partial u_i}{\partial t}-\frac{1}{\rho}\frac{\partial p}{\partial x_i}+\nu\frac{\partial}{\partial x_j}\left(\frac{\partial u_j}{\partial x_i}+\frac{\partial u_i}{\partial x_j}\right)\right\}\mathrm{d}\Omega\,\mathrm{d}t=0\\
&i,j,k=1,2,3\in\mathbf{R}^1\quad\text{in }\Omega
\end{aligned}
\tag{B3}
$$

A time t and domain Ω integral term above the equation satisfies Equations (23), (33), (39), (38), (17), (18), (19), and (20) over Ω. This part term in the above equation is as follows:

$$\int_{t_\mathrm{s}}^{t_\mathrm{e}}\int_{\Omega(\mathbf{S})}\frac{\partial p}{\partial x_k}\left(\frac{\partial\lambda_2}{\partial x}+\frac{\partial\lambda_3}{\partial y}+\frac{\partial\lambda_4}{\partial z}\right)\frac{1}{\rho}\mathrm{d}\Omega\,\mathrm{d}t=0\in\mathbf{R}^1\quad\text{in }\Omega \tag{B4}$$

$$\int_{t_\mathrm{s}}^{t_\mathrm{e}}\int_{\Omega(\mathbf{S})}\frac{\partial u_i}{\partial x_k}\left\{\frac{\partial\lambda_{i+1}}{\partial t}-\frac{1}{\rho}\frac{\partial\lambda_1}{\partial x_i}+\nu\frac{\partial}{\partial x_j}\left(\frac{\partial\lambda_{j+1}}{\partial x_i}+\frac{\partial\lambda_{i+1}}{\partial x_j}\right)\right\}\mathrm{d}\Omega\,\mathrm{d}t=0\in\mathbf{R}^1\quad\text{in }\Omega \tag{B5}$$

$$\int_{t_\mathrm{s}}^{t_\mathrm{e}}\int_{\Omega(\mathbf{S})}\left(\frac{\partial u}{\partial x}+\frac{\partial v}{\partial y}+\frac{\partial w}{\partial z}\right)\frac{\partial\lambda_1}{\partial x_k}\frac{1}{\rho}\mathrm{d}\Omega\,\mathrm{d}t=0\in\mathbf{R}^1\quad\text{in }\Omega \tag{B6}$$

$$\int_{t_\mathrm{s}}^{t_\mathrm{e}}\int_{\Omega(\mathbf{S})}\frac{\partial\lambda_{i+1}}{\partial x_k}\left\{-\frac{\partial u_i}{\partial t}-\frac{1}{\rho}\frac{\partial p}{\partial x_i}+\nu\frac{\partial}{\partial x_j}\left(\frac{\partial u_j}{\partial x_i}+\frac{\partial u_i}{\partial x_j}\right)\right\}\mathrm{d}\Omega\,\mathrm{d}t=0\in\mathbf{R}^1\quad\text{in }\Omega \tag{B7}$$

Since Ω term in Equation (B3) should be satisfied at arbitrary space, this term should be equal to zero. Thus, we obtain

$$\lambda_{i+1}(t_\mathrm{s},\mathbf{X})\frac{\partial u_i(t_\mathrm{s},\mathbf{X})}{\partial x_k}-\lambda_{i+1}(t_\mathrm{e},\mathbf{X})\frac{\partial u_i(t_\mathrm{e},\mathbf{X})}{\partial x_k}=0\in\mathbf{R}^1\quad\text{in }\Omega \tag{B8}$$

By using Equations (30), (35), (38), $\lambda_{i+1}(t_\mathrm{s},\mathbf{X})\neq 0$, and $\lambda_{i+1}(t_\mathrm{e},\mathbf{X})\neq 0$, we obtain

$$\frac{\partial u_i(t_\mathrm{s},\mathbf{X})}{\partial x_k}-\frac{\partial u_i(t_\mathrm{e},\mathbf{X})}{\partial x_k}=0\in\mathbf{R}^1\quad\text{in }\Omega \tag{B9}$$

In order to satisfy the above equation, the value of the state variable at the end time t_e must equal the value at the start time t_s in the computational domain Ω. The state variable as the start time and the end time is set as:

$$u_i(t_\mathrm{s},\mathbf{X})-u_i(t_\mathrm{e},\mathbf{X})=0\in\mathbf{R}^1\quad\text{in }\Omega \tag{B10}$$

A boundary Ψ term in the above equation is as follows:

$$
\begin{aligned}
&\lambda_{i+1}\left\{-\frac{1}{\rho}\frac{\partial p}{\partial x_k}n_i+v\frac{\partial}{\partial x_k}\left(\frac{\partial u_i}{\partial x_j}+\frac{\partial u_j}{\partial x_i}\right)n_j\right\}+\lambda_{i+1}\left\{-\frac{p}{\rho}\frac{\partial n_i}{\partial x_k}+v\left(\frac{\partial u_i}{\partial x_j}+\frac{\partial u_j}{\partial x_i}\right)\frac{\partial n_j}{\partial x_k}\right\}\\
&-\frac{\partial u_i}{\partial x_k}\left\{-\frac{\lambda_1}{\rho}n_i+v\left(\frac{\partial \lambda_{i+1}}{\partial x_j}+\frac{\partial \lambda_{j+1}}{\partial x_i}\right)n_j\right\}\\
&-u_i\left\{-\frac{\lambda_1}{\rho}\frac{\partial n_i}{\partial x_k}+v\left(\frac{\partial \lambda_{i+1}}{\partial x_j}+\frac{\partial \lambda_{j+1}}{\partial x_i}\right)\frac{\partial n_j}{\partial x_k}\right\}=0 \quad \text{on } \Gamma
\end{aligned}
\tag{B11}
$$

For the boundary Γ_w, by using the normal $(n_1 \;\; n_2 \;\; n_3)=(1 \;\; 0 \;\; 0)$, we obtain

$$
\left(\frac{\partial n_1}{\partial x_k}\frac{\partial n_2}{\partial x_k}\frac{\partial n_3}{\partial x_k}\right)=(0 \;\; 0 \;\; 0) \quad \text{on } \Gamma_{\mathrm{W}}
\tag{B12}
$$

By using Equation (27), we obtain

$$
\lambda_{i+1}\left\{-\frac{\partial p}{\partial x_k}n_i+\mu\frac{\partial}{\partial x_k}\left(\frac{\partial u_i}{\partial x_j}+\frac{\partial u_j}{\partial x_i}\right)n_j\right\}=0 \quad \text{on } \Gamma_{\mathrm{W}}
\tag{B13}
$$

By using Equations (29), (34) and (37), we obtain

$$
-\frac{\partial u_i}{\partial x_k}\left\{-\lambda_1 n_i+\mu\left(\frac{\partial \lambda_{i+1}}{\partial x_j}+\frac{\partial \lambda_{j+1}}{\partial x_i}\right)n_j\right\}=0 \quad \text{on } \Gamma_{\mathrm{W}}
\tag{B14}
$$

By using Equations (B12)–(B14), we understand that Equation (B11) is satisfied. The other boundaries are also satisfied in a similar way.

The boundary γ term in Equation (B3) and the definition Equation (1) are as follows:

$$
\begin{aligned}
&-\int_{t_{\mathrm{s}}}^{t_{\mathrm{e}}}\int_{\gamma(\mathbf{S})}\left\{-\frac{1}{\rho}\frac{\partial p}{\partial x_k}n_1+v\frac{\partial}{\partial x_k}\left(\frac{\partial u}{\partial x_j}+\frac{\partial u_j}{\partial x}\right)n_j\right\}\mathrm{d}\gamma\,\mathrm{d}t\\
&-\int_{t_{\mathrm{s}}}^{t_{\mathrm{e}}}\int_{\gamma(\mathbf{S})}\left\{-\frac{p}{\rho}\frac{\partial n_1}{\partial x_k}+v\left(\frac{\partial u}{\partial x_j}+\frac{\partial u_j}{\partial x}\right)\frac{\partial n_j}{\partial x_k}\right\}\mathrm{d}\gamma\,\mathrm{d}t\\
&+\int_{t_{\mathrm{s}}}^{t_{\mathrm{e}}}\int_{\gamma(\mathbf{S})}\lambda_{i+1}\left\{-\frac{1}{\rho}\frac{\partial p}{\partial x_k}n_i+v\frac{\partial}{\partial x_k}\left(\frac{\partial u_i}{\partial x_j}+\frac{\partial u_j}{\partial x_i}\right)n_j\right\}\mathrm{d}\gamma\,\mathrm{d}t\\
&+\int_{t_{\mathrm{s}}}^{t_{\mathrm{e}}}\int_{\gamma(\mathbf{S})}\lambda_{i+1}\left\{-\frac{p}{\rho}\frac{\partial n_i}{\partial x_k}+v\left(\frac{\partial u_i}{\partial x_j}+\frac{\partial u_j}{\partial x_i}\right)\frac{\partial n_j}{\partial x_k}\right\}\mathrm{d}\gamma\,\mathrm{d}t\\
&-\int_{t_{\mathrm{s}}}^{t_{\mathrm{e}}}\int_{\gamma(\mathbf{S})}\frac{\partial u_i}{\partial x_k}\left\{-\frac{\lambda_1}{\rho}n_i+v\left(\frac{\partial \lambda_{i+1}}{\partial x_j}+\frac{\partial \lambda_{j+1}}{\partial x_i}\right)n_j\right\}\mathrm{d}\gamma\,\mathrm{d}t\\
&-\int_{t_{\mathrm{s}}}^{t_{\mathrm{e}}}\int_{\gamma(\mathbf{S})}u_i\left\{-\frac{\lambda_1}{\rho}\frac{\partial n_i}{\partial x_k}+v\left(\frac{\partial \lambda_{i+1}}{\partial x_j}+\frac{\partial \lambda_{j+1}}{\partial x_i}\right)\frac{\partial n_j}{\partial x_k}\right\}\mathrm{d}\gamma\,\mathrm{d}t=0\\
&i,j,k=1,2,3
\end{aligned}
\tag{B15}
$$

By using the boundary condition $\lambda=(1,0,0), u=(0,0,0)$ on the surface γ, Equation (B15) is as follows:

$$\begin{aligned}
&-\int_{t_\mathrm{s}}^{t_\mathrm{e}}\int_{\gamma(\mathbf{S})}\left\{-\frac{1}{\rho}\frac{\partial p}{\partial x_k}n_1+v\frac{\partial}{\partial x_k}\left(\frac{\partial u}{\partial x_j}+\frac{\partial u_j}{\partial x}\right)n_j\right\}\mathrm{d}\gamma\,\mathrm{d}t\\
&-\int_{t_\mathrm{s}}^{t_\mathrm{e}}\int_{\gamma(\mathbf{S})}\left\{-\frac{p}{\rho}\frac{\partial n_1}{\partial x_k}+v\left(\frac{\partial u}{\partial x_j}+\frac{\partial u_j}{\partial x}\right)\frac{\partial n_j}{\partial x_k}\right\}\mathrm{d}\gamma\,\mathrm{d}t\\
&+\int_{t_\mathrm{s}}^{t_\mathrm{e}}\int_{\gamma(\mathbf{S})}\left\{-\frac{p}{\rho}\frac{\partial n_1}{\partial x_k}+v\left(\frac{\partial u}{\partial x_j}+\frac{\partial u_j}{\partial x}\right)\frac{\partial n_j}{\partial x_k}\right\}\mathrm{d}\gamma\,\mathrm{d}t\\
&+\int_{t_\mathrm{s}}^{t_\mathrm{e}}\int_{\gamma(\mathbf{S})}\left\{-\frac{p}{\rho}\frac{\partial n_1}{\partial x_k}+v\left(\frac{\partial u}{\partial x_j}+\frac{\partial u_j}{\partial x}\right)\frac{\partial n_j}{\partial x_k}\right\}\mathrm{d}\gamma\,\mathrm{d}t\\
&-\int_{t_\mathrm{s}}^{t_\mathrm{e}}\int_{\gamma(\mathbf{S})}\frac{\partial u_i}{\partial x_k}\left\{-\frac{\lambda_1}{\rho}n_i+v\left(\frac{\partial \lambda_{i+1}}{\partial x_j}+\frac{\partial \lambda_{j+1}}{\partial x_i}\right)n_j\right\}\mathrm{d}\gamma\,\mathrm{d}t=0\\
&i,j,k=1,2,3
\end{aligned}\tag{B16}$$

In this study, the domain of optimization is the boundary γ. The boundary γ satisfies the non-slip condition of the boundary γ (Table I) and Equation (25). By using these conditions, Equation (B16) is transformed as follows:

$$\frac{\partial u_i}{\partial x_k}\left\{-\lambda_1 n_i+\mu\left(\frac{\partial \lambda_{i+1}}{\partial x_j}+\frac{\partial \lambda_{j+1}}{\partial x_i}\right)n_j\right\}=0,\quad i,j,k=1,2,3\quad \text{on } \gamma \tag{B17}$$

For the above equations, the sensitivity equations (Equations (41)–(43)) are derived. Boundary conditions Ψ can be assumed to satisfy Equations (29), (34), and (37) by using $\lambda_1=0$ and Equation (27) as shown in Table I, while the boundary γ (the surface) cannot be assumed to satisfy Equations (29), (34), and (37) because traction on the boundary γ is generated by the adjoint flow.

ACKNOWLEDGEMENTS

This work was supported by the JAEA (Japan Atomic Energy Agency) and the IML (Intelligent Modeling Laboratory). The first author is grateful to JAEA for the financial support as a research student and to the IML for the financial support as a postdoctoral fellow. This work was also supported by the HEC-MW group (Frontier Simulation Software for Industrial Science (FSIS) project, which started in 2002 and has been driven for more than three years, has developed and released more than 60 pieces of most advanced simulation software in the field of computational mechanics). This program, based on HEC-MW, was improved by useful advice from a large number of people, who also helped debug it.

REFERENCES

1. Jameson A. Aerodynamic design via control theory. *Journal of Scientific Computing* 1988; **3**:233–260.
2. Ragab SA. Shape optimization of surface ships in potential flow using an adjoint formulation. *AIAA Journal* 2004; **42**:296–304.
3. Campobasso AS, Duta MC, Giles MB. Adjoint calculation of sensitivities of turbomachinery objective functions. *Journal of Propulsion and Power* 2003; **19**:693–703.

4. Bangtsson E, Noreland D, Berggren M. Shape optimization of an acoustic horn. *Computer Methods in Applied Mechanics and Engineering* 2003; **192**:1533–1571.
5. Yamaguchi Y, Ogura D, Yamashita K, Miyazaki M, Nakamura H, Maeda H. A method for DNA detection in a microchannel fluid dynamics phenomena and optimization of microchannel structure. *Talanta* 2006; **68**: 700–707.
6. Bull JL, Hunt AJ, Meyhofer E. A theoretical model of a molecular-motor-powered pump. *Biomedical Microdevices* 2005; **7**:21–33.
7. Burgreen GW, Antaki JF, Griffith BP. CFD-based design optimization of a three-dimensional rotary blood pump. *AIAA Journal* 1996; 96–4185.
8. Li HB, Bashir R. On the design and optimization of microfluidic dielectrophoretic devices: a dynamic simulation study. *Biomedical Microdevices* 2004; **6**:289–295.
9. Chen CF, Kung CF, Chen HC, Chu CC, Chang CC, Tseng FG. A microfluidic nanoliter mixer with optimized grooved structures driven by capillary pumping. *Journal of Micromechanics and Microengineering* 2006; **16**: 1358–1365.
10. Eijkel J. Liquid slip in micro- and nanofluidics: recent research and its possible implications. *Lab on a Chip* 2007; **7**:299–301.
11. Foley JO, Nelson KE, Mashadi-Hossein A, Finlayson BA, Yager P. Concentration gradient immunoassay 2. Computational modeling for analysis and optimization. *Analytical Chemistry* 2007; **79**:3549–3553.
12. Pironneau O. *Optimal Shape Design for Elliptic Systems*. Springer: New York, 1984.
13. Sano M, Sakai H. Numerical determination of minimum drag profile in stokes flow (in case of two-dimensional finite region and constant cross-sectional area). *Journal of the Japan Society for Aeronautical and Space Sciences* 1983; **26**:1–9.
14. Bessho M, Himeno Y. On optimum profiles in two-dimensional stokes flow. *The Japan Society of Naval Architects and Ocean Engineers Kansai Division* 1983; **193**:115–125.
15. Taseli H, Demiralp M. Drag minimization in stokes flow. *International Journal of Engineering Science* 1989; **27**:633–640.
16. Ganesh RK. The minimum drag profile in laminar flow. *Journal of Fluids Engineering* 1994; **116**:456–462.
17. Huan J, Modi V. Design of minimum drag bodies in incompressible laminar flow. *Inverse Problems in Engineering* 1996; **3**:233–260.
18. Schweyher H, Lutz Th, Wagner S. An optimization tool for axisymmetric bodies of minimum drag. *Proceedings of 2nd International Airship Conference*, Stuttgart/Friedrichshafen, Germany, 1996; 3–4.
19. Katamine E, Azegami H. Solution to viscous flow field domain optimization problems. *The Japan Society of Mechanical Engineers* 1994; **60**:1479–1486.
20. Richardson S. Optimum profiles in two-dimensional Stokes flow. *Proceedings of the Royal Society of London Series A—Mathematical and Physical Sciences* 1995; **450**:603–622.
21. Kim DW, Kim MU. Minimum drag shape in 2-dimensional viscous-flow. *International Journal for Numerical Methods in Fluids* 1995; **21**:93–111.
22. Datta S, Srivastava DK. Optimum drag profile in axisymmetric stokes flow. *Indian Journal of Pure and Applied Mathematics* 2002; **33**:409–426.
23. Lund E, Moller H, Jakobsen LA. Shape design optimization of stationary fluid–structure interaction problems with large displacements and turbulence. *Structural and Multidisciplinary Optimization* 2003; **25**:383–392.
24. Yagi H, Kawahara M. Shape optimization of a body located in low Reynolds number flow. *International Journal for Numerical Methods in Fluids* 2005; **48**:819–833.
25. Matsumoto J. Shape identification for Navier–Stokes equation with unsteady flow using bubble function element stabilization method. *World Congress on Computational Mechanics*, Beijing, China, vol. 353, 2004.
26. Guest JK, Prevost JH. Topology optimization of creeping fluid flows using a darcy-stokes finite element. *International Journal for Numerical Methods in Engineering* 2006; **66**:461–484.
27. Mavriplis DJ. Multigrid solution of the discrete adjoint for optimization problems on unstructured meshes. *AIAA Journal* 2006; **44**:42–50.
28. HEC-MW. http://www.rss21.iis.u-tokyo.ac.jp/en/result/download/.
29. Gilboa G, Sochen N, Zeevi Y. Forward-and-backward diffusion processes for adaptive image enhancement and denoising. *IEEE Transaction on Image Processing* 2002; **11**:689–703.
30. Kirkup SM, Wadsworth M. Solution of inverse diffusion problems by operator-splitting methods. *Applied Mathematical Modeling* 2002; **26**:1003–1018.
31. Laporte E, Tallec P. *Numerical Methods in Sensitivity Analysis and Shape Optimization*. Birkhauser: Basel, 2003; 127.

DOI: 10.1002/fld

32. Soto O, Lohner R. A mixed adjoint formulation for incompressible turbulent problems. *AIAA Journal* 2002; **0451**.
33. Kim HJ, Salim K, Nakahashi K. Surface modification method for aerodynamic design optimization. *AIAA Journal* 2005; **43**:727–740.
34. Canann SA, Liu YC, Mobley AV. Automatic 3D surface meshing to address today's industrial needs. *Finite Elements in Analysis and Design* 1997; **25**:185–198.
35. Canann SA, Tristano JR, Staten ML. An approach to combined Laplacian and optimization-based smoothing for triangular quadrilateral and quad-dominant meshes. *Seventh International Meshing Roundtable*, Dearborn, U.S.A., 1998; 479–494.
36. Freitag LA, Knupp PM. Tetrahedral mesh improvement via optimization of the element condition number. *International Journal for Numerical Methods in Engineering* 2002; **53**:1377–1391.
37. Shinohara K, Okuda H, Ito S, Nakajima N, Ida M. Shape optimization using an adjont variable method in ITBL grid environment. *Fourteenth International Conference on Nuclear Engineering*, Miami, U.S.A., 2006; **14**:89568.
38. Brian TH. Mesh deformation using the biharmonic operator. *International Journal for Numerical Methods in Engineering* 2003; **56**:1007–1021.
39. Saleh JM. *Fluid Flow Handbook*. McGraw-Hill: New York, 2002.

DOI: 10.1002/fld