10th World Congress on Structural and Multidisciplinary Optimization
May 19 - 24, 2013, Orlando, Florida, USA

Level-set and ALE Based Topology Optimization Using Nonlinear Programming

Shintaro Yamasaki[1], Atsushi Kawamoto[2] and Tsuyoshi Nomura[3]

[1]Department of Mechanical Engineering, Osaka University, Suita, Japan, yamasaki@mech.eng.osaka-u.ac.jp
[2]Vehicle Safety Research Center, Toyota Central R&D Labs., Inc., Nagakute, Japan, atskwmt@mosk.tytlabs.co.jp
[3]Electronics Research Department, Toyota Research Institute of North America, Inc., MI, USA, tsuyoshi.nomura@tema.toyota.com

1. Abstract
In this paper, we propose a topology optimization method based on the combination of the level-set representation and the arbitrary Lagrangian Eulerian (ALE) moving mesh, where the level-set function is updated by nonlinear programming. The major advantage of this method is that grayscale elements are completely eliminated by providing a mesh conformal with the level-set boundary. Previously, we introduced a level-set based topology optimization method with the ALE moving mesh, but the level-set function was updated by solving the level-set equation. In order to handle multiple constraints, we adopt the method to update the level-set function using mathematical programming method with exact sensitivities. We explain the details of the proposed method and demonstrate the usefulness of the proposed method using a numerical example in this paper.
2. Keywords: Level-set Method, Arbitrary Lagrangian Eulerian Method, Topology Optimization, Nonlinear Programming.

3. Introduction
Topology optimization [1, 2] is a powerful design approach with its capability to provide superior designs while considering both shape and topology simultaneously. Level-set based topology optimization [3, 4], one variety of the topology optimization method, is nowadays well studied because of its potential capability that stems from the level-set method [5].

The level-set method is a shape representation method on the Eulerian coordinate system. In this method, a scalar function called the level-set function is used to represent the shape and topology of two phases, e.g., Phase 1 and 2. When the value of the level-set function is positive at a point, the point is regarded as in Phase 1 domain. On the other hand, when the value is negative, the point is regarded as in Phase 2 domain. The zero iso-contour of the level-set function represents the boundary between the two phases. Therefore, the shape boundary of the target structure is always clearly represented during optimization by applying the level-set method to structural topology optimization. Furthermore, updating the level-set function allows various topological changes of the target structure.

For the numerical analysis, the design domain of level-set based topology optimization is discretized with the Eulerian mesh, and the level-set function is maintained in the Eulerian mesh. In usual cases, the shape boundaries of the target structure are not coincident with the Eulerian mesh, so, some numerical workarounds are required to compute the state variables. One typical numerical technique is to allow intermediate material property around the shape boundaries. This technique yields so-called grayscale elements around the shape boundary. In this manner, the state variables can be computed using the Eulerian mesh without an additional mesh for computing the state variables. However, the grayscale elements are not preferred in the engineering view point due to its difficulty of interpretation of their material status.

To avoid this problem, we have proposed a level-set based topology optimization method where the arbitrary Lagrangian Eulerian (ALE) method is incorporated [6]. In this method, we provide two types of mesh for level-set based topology optimization; one is the Eulerian mesh that maintains the level-set function, and the other is the ALE mesh for computing the state variables. Since the ALE mesh accurately tracks the shape boundary during optimization on the basis of the ALE method, grayscale elements can be completely eliminated. The proposed method was applied to metallic waveguide design problems [6] and a compliant mechanism design problem [7], and reasonable designs were obtained in these previous studies.

A simple scheme based on the level-set equation [8] was adopted for updating the level-set function in studies [6, 7]. In this paper, the level-set function is updated using nonlinear programming. By doing this, handling of multiple nonlinear constraints is expected to be improved. We demonstrate the usefulness of the newly proposed method by solving a compliant mechanism design problem.

The rest of this paper is organized as follows. We formulate the compliant mechanism design problem using the level-set method in Section 4. Further, we explain the numerical implementation of the newly proposed method in Section 5. In Section 6, we provide a numerical example to demonstrate the usefulness of the proposed method. Finally, we conclude the discussion in Section 7.

4. Formulation

4.1. Shape representation based on the level-set method

In this section, we briefly explain the shape representation based on the level-set method. Consider the structural design in a given design domain D consisting of a type of structural material and air. The material domain, the domain filled with the material, and the void domain, the domain filled with air, are respectively denoted as Ω and $D \setminus \Omega$. Then, arbitrary shape and topology of Ω in D is represented using the level-set function ϕ as follows:

$$\begin{aligned} \phi(\mathbf{x}) > 0 \quad & \text{for} \quad \forall \mathbf{x} \in \Omega \setminus (\partial\Omega \setminus \partial D), \\ \phi(\mathbf{x}) = 0 \quad & \text{for} \quad \forall \mathbf{x} \in \partial\Omega \setminus \partial D, \\ \phi(\mathbf{x}) < 0 \quad & \text{for} \quad \forall \mathbf{x} \in D \setminus \Omega, \end{aligned} \tag{1}$$

where $\mathbf{x}$ is the coordinates of a point in D, ∂D is the boundary of D, and $\partial\Omega$ is the boundary of Ω.

The structure represented by Ω is the target structure of level-set based topology optimization, and the target structure is profiled by the zero iso-contour of the level set function, i.e., the level-set boundary.

4.2. Compliant mechanism design problem

We formulate a compliant mechanism design problem with the level-set based shape representation. First, a domain having an input port ∂D_{in} and an output port ∂D_{out} is provided as the design domain D. As explained in Section 4.1, the design domain D consists of the material domain Ω and the void domain $D \setminus \Omega$. Since the material in Ω has an appropriate stiffness, the target structure is deformed by an input load on ∂D_{in}. Our purpose is to maximize a specified direction deformation on ∂D_{out} under three constraints. The first constraint is the volume constraint that limits material usage. The second constraint is a compliance constraint that prevents an extreme deformation by the input load. The third constraint is another compliance constraint to conduct enough force to the work piece set on the output port. The design problem is formulated as follows:

$$\begin{aligned} \text{minimize} \quad & f_0(\phi) := -\int_{\partial D_{\text{out}}} \mathbf{t}_{\text{out}} \cdot \mathbf{u}_1 \mathrm{d}s, \\ \text{subject to} \quad & f_1(\phi) := \int_D H(\phi(\mathbf{x})) \mathrm{d}v \leq f_{1\max}, \\ & f_2(\phi) := \int_{\partial D_{\text{in}}} \mathbf{t}_{\text{in}} \cdot \mathbf{u}_1 \mathrm{d}s \leq f_{2\max}, \\ & f_3(\phi) := \int_{\partial D_{\text{out}}} \mathbf{t}_{\text{out}} \cdot \mathbf{u}_2 \mathrm{d}s \leq f_{3\max}, \end{aligned} \tag{2}$$

where $\mathbf{t}_{\text{in}}$ is the input load, and $\mathbf{t}_{\text{out}}$ is the dummy output load that specifies the desired deformation direction on the output port. $f_{1\max}$, $f_{2\max}$, and $f_{3\max}$ are the allowable maximum value of $f_1(\phi)$, $f_2(\phi)$, and $f_3(\phi)$, respectively. $H(\phi(\mathbf{x}))$ is the Heaviside function and it is defined as

$$H(\phi(\mathbf{x})) = \begin{cases} 0 & \text{if} \quad \phi(\mathbf{x}) < 0, \\ 1 & \text{otherwise.} \end{cases} \tag{3}$$

$\mathbf{u}_1$ is the displacement field when $\mathbf{t}_{\text{in}}$ is applied on ∂D_{in}. It is obtained by solving the following equation:

$$\int_{\partial D_{\text{in}}} \mathbf{t}_{\text{in}} \cdot \hat{\mathbf{u}}_1 \mathrm{d}s = \int_D \boldsymbol{\epsilon}(\hat{\mathbf{u}}_1) : \mathbf{D} : \boldsymbol{\epsilon}(\mathbf{u}_1) H(\phi(\mathbf{x})) \mathrm{d}v, \tag{4}$$

where $\hat{\mathbf{u}}_1$ is the test function of $\mathbf{u}_1$, $\boldsymbol{\epsilon}$ is the linearized strain tensor, and $\mathbf{D}$ is the isotropic elasticity tensor. Similarly, $\mathbf{u}_2$ is the displacement field when $\mathbf{t}_{\text{out}}$ is applied on ∂D_{out}, and it is obtained by

$$\int_{\partial D_{\text{out}}} \mathbf{t}_{\text{out}} \cdot \hat{\mathbf{u}}_2 \mathrm{d}s = \int_D \boldsymbol{\epsilon}(\hat{\mathbf{u}}_2) : \mathbf{D} : \boldsymbol{\epsilon}(\mathbf{u}_2) H(\phi(\mathbf{x})) \mathrm{d}v, \tag{5}$$

where $\hat{\mathbf{u}}_2$ is the test function of $\mathbf{u}_2$.

5. Implementation

5.1. State variable computation using the ALE mesh

In this paper, we use the method proposed in our previous paper [7] to compute the state variables $\mathbf{u}_1$ and $\mathbf{u}_2$. In this method, the level-set function is maintained in the Eulerian mesh, whereas the state variables are computed using the ALE mesh that accurately tracks the level-set boundary. The ALE mesh is provided at every optimization iteration in the manner explained in the paper [7].

Equations (4) and (5) are discretized using the ALE mesh as follows:

$$\hat{\mathbf{U}}_1^\top \{\mathbf{T}_{\text{in}} - \mathbf{K}\mathbf{U}_1\} = 0, \tag{6}$$

$$\hat{\mathbf{U}}_2^\top \{\mathbf{T}_{\text{out}} - \mathbf{K}\mathbf{U}_2\} = 0, \tag{7}$$

where $\mathbf{T}_{\text{in}}$ and $\mathbf{T}_{\text{out}}$ are the discretized loads of $\mathbf{t}_{\text{in}}$ and $\mathbf{t}_{\text{out}}$, respectively, $\mathbf{U}_1$ and $\mathbf{U}_2$ are the nodal value vectors of $\mathbf{u}_1$ and $\mathbf{u}_2$, respectively, and $\hat{\mathbf{U}}_1$ and $\hat{\mathbf{U}}_2$ are the discretized test functions of $\hat{\mathbf{u}}_1$ and $\hat{\mathbf{u}}_2$, respectively. $\mathbf{K}$ is the total stiffness matrix given as

$$\mathbf{K} = \bigcup_{e=1}^{n_\Omega} \int_{V^{\text{e}}} \mathbf{B}^\top \mathbf{D}^{\text{H}} \mathbf{B} \mathrm{d}v + \bigcup_{e=1}^{n_{D\setminus\Omega}} \int_{V^{\text{e}}} \mathbf{B}^\top \kappa \mathbf{D}^{\text{H}} \mathbf{B} \mathrm{d}v, \tag{8}$$

where $\int_{V^{\text{e}}} \mathrm{d}v$ represents the volume integration of an element, $\mathbf{B}$ is the matrix representing the relation between the nodal displacement and the strain in an element, $\mathbf{D}^{\text{H}}$ is the elasticity tensor written in the matrix form, and κ is an infinitesimal value for avoiding the singularity of $\mathbf{K}$. $\bigcup_{e=1}^{n_\Omega}$ and $\bigcup_{e=1}^{n_{D\setminus\Omega}}$ represent the union sets of the elements in Ω and $D \setminus \Omega$, respectively.

5.2. Sensitivities with respect to the nodal coordinates of the ALE mesh

In this section, we derive the sensitivities of the objective and constraint functions with respect to the nodal coordinates on the level-set boundary in the ALE mesh. Since $\mathbf{u}_1$ is approximately computed using the ALE mesh, the object function $f_0(\phi)$ is also approximately computed using the ALE mesh. We denote the approximately computed $f_0(\phi)$ as $\tilde{f}_0$, then the following is obtained from Equation (2):

$$\tilde{f}_0 = -{\mathbf{U}_1}^\top \mathbf{T}_{\text{out}}. \tag{9}$$

Furthermore, $\tilde{f}_0$ is described as follows because of Equations (6) and (9):

$$\tilde{f}_0 = -{\mathbf{U}_1}^\top \mathbf{T}_{\text{out}} + \mathbf{W}^\top \{\mathbf{T}_{\text{in}} - \mathbf{K}\mathbf{U}_1\}, \tag{10}$$

where $\mathbf{W}$ is the nodal adjoint variable vector.

Here, we consider the variation of $\tilde{f}_0$ with respect to the coordinate variation of a node on the level-set boundary, and denote it as $\delta\tilde{f}_0$. From Equation (10),

$$\delta\tilde{f}_0 = -{\mathbf{U}_1}^\top \delta\mathbf{T}_{\text{out}} + \mathbf{W}^\top \{\delta\mathbf{T}_{\text{in}} - \delta\mathbf{K}\mathbf{U}_1\} - \delta{\mathbf{U}_1}^\top \{\mathbf{T}_{\text{out}} + \mathbf{K}^\top \mathbf{W}\}, \tag{11}$$

where $\delta\mathbf{T}_{\text{in}}$, $\delta\mathbf{T}_{\text{out}}$, $\delta\mathbf{K}$, and $\delta\mathbf{U}_1$ are, respectively, the variations of $\mathbf{T}_{\text{in}}$, $\mathbf{T}_{\text{out}}$, $\mathbf{K}$, and $\mathbf{U}_1$. To simplify the derivation of the sensitivity, we assume that the level-set boundary does not contact ∂D_{in} and ∂D_{out}. Under this assumption, $\delta\mathbf{T}_{\text{in}}$ and $\delta\mathbf{T}_{\text{out}}$ can be ignored, therefore,

$$\delta\tilde{f}_0 = -\mathbf{W}^\top \delta\mathbf{K}\mathbf{U}_1 - \delta{\mathbf{U}_1}^\top \{\mathbf{T}_{\text{out}} + \mathbf{K}^\top \mathbf{W}\}. \tag{12}$$

Since the second term of the right hand side of the above equation is deleted by substituting $\mathbf{W} = -\mathbf{U}_2$,

$$\delta\tilde{f}_0 = {\mathbf{U}_2}^\top \delta\mathbf{K}\mathbf{U}_1. \tag{13}$$

Therefore, the nodal coordinate sensitivities on the level-set boundary are obtained as follows:

$$\frac{\partial \tilde{f}_0}{\partial X} = {\mathbf{U}_2}^\top \frac{\partial \mathbf{K}}{\partial X}\mathbf{U}_1, \qquad \frac{\partial \tilde{f}_0}{\partial Y} = {\mathbf{U}_2}^\top \frac{\partial \mathbf{K}}{\partial Y}\mathbf{U}_1, \tag{14}$$

where X and Y are, respectively, the x and y coordinates of a node on the level-set boundary.

Next, we consider the sensitivities of the constraint functions. We denote the approximately computed $f_1(\phi)$, $f_2(\phi)$, and $f_3(\phi)$ as $\tilde{f}_1$, $\tilde{f}_2$, and $\tilde{f}_3$, respectively. These are computed as

$$\tilde{f}_1 = \bigcup_{e=1}^{n_\Omega} \int_{V^e} \mathrm{d}v, \tag{15}$$

$$\tilde{f}_2 = {\mathbf{U}_1}^\top \mathbf{T}_{\mathrm{in}}, \tag{16}$$

$$\tilde{f}_3 = {\mathbf{U}_2}^\top \mathbf{T}_{\mathrm{out}}, \tag{17}$$

in the discrete system. The sensitivity of $\tilde{f}_1$ is derived as

$$\frac{\partial \tilde{f}_1}{\partial X} = \frac{\mathrm{d}}{\mathrm{d}X}\Big\{\bigcup_{e=1}^{n_\Omega} \int_{V^e} \mathrm{d}v\Big\}, \qquad \frac{\partial \tilde{f}_1}{\partial Y} = \frac{\mathrm{d}}{\mathrm{d}Y}\Big\{\bigcup_{e=1}^{n_\Omega} \int_{V^e} \mathrm{d}v\Big\}. \tag{18}$$

The sensitivities of $\tilde{f}_2$ and $\tilde{f}_3$ are derived as follows similarly to $\tilde{f}_0$:

$$\frac{\partial \tilde{f}_2}{\partial X} = -{\mathbf{U}_1}^\top \frac{\partial \mathbf{K}}{\partial X}\mathbf{U}_1, \qquad \frac{\partial \tilde{f}_2}{\partial Y} = -{\mathbf{U}_1}^\top \frac{\partial \mathbf{K}}{\partial Y}\mathbf{U}_1, \tag{19}$$

$$\frac{\partial \tilde{f}_3}{\partial X} = -{\mathbf{U}_2}^\top \frac{\partial \mathbf{K}}{\partial X}\mathbf{U}_2, \qquad \frac{\partial \tilde{f}_3}{\partial Y} = -{\mathbf{U}_2}^\top \frac{\partial \mathbf{K}}{\partial Y}\mathbf{U}_2. \tag{20}$$

5.3. Sensitivities with respect to the nodal level-set function of the Eulerian mesh

In Section 5.2, we derived the sensitivities of the objective and constraint functions with respect to the nodal coordinates on the level-set boundary in the ALE mesh. In the previous work [7], this kind of sensitivities were smoothed using a Helmholtz-type PDE [9, 10], and the smoothed sensitivities were used to compute the normal velocity of the level-set equation, and the level-set function was iteratively updated by solving the level-set equation in the Eulerian mesh. However, we propose to update the level-set function using nonlinear programming in this paper, because it is reasonable to use nonlinear programming when multiple nonlinear constraints are imposed. For this purpose, we derive the sensitivities with respect to the nodal level-set function of the Eulerian mesh.

Here, we explain the relationship between the Eulerian mesh and the level-set boundary of the ALE mesh. Figure 1(a) is an example of the Eulerian mesh for the explanation. As shown in Figure 1(a), the design domain is discretized using triangular elements. Since these triangular elements are linear elements, the level-set function is maintained at the vertex nodes of the triangular elements. The shape and topology of the material domain are given by the level-set function values maintained at each node. In the case of Figure 1(a), we assume that the material domain is distributed as displayed in gray. Then, the corresponding ALE mesh is obtained as shown in Figure 1(b) by the ALE mesh generating scheme proposed in study [6]. In this figure, the level-set boundary is represented by the thick dotted lines (over the thin solid lines), and the nodes on the level-set boundary are represented by the red nodes. Figure 1(c) shows the superposition of the Eulerian mesh and the level-set boundary.

Since the sensitivities with respect to the red node coordinates can be derived as discussed in Section 5.2, we can derive the sensitivities with respect to the nodal level-set function of the Eulerian mesh if we derive the sensitivities of the red node coordinates with respect to the nodal level-set function of the Eulerian mesh. Therefore, we focus on a triangular element in Figure 1(c) which include a red node, and show this element in Figure 1(d). We denote the coordinates of Nodes 1, 2, and 3 of Figure 1(d) as (x_1, y_1), (x_2, y_2), and (x_3, y_3), respectively. Similarly, we denote these nodal level-set function as ϕ_1, ϕ_2, and ϕ_3, respectively. Since the red node is on the line segment where $\phi = 0$, the coordinates of the red node, (X, Y), are then computed as follows:

$$\begin{bmatrix} X \\ Y \end{bmatrix} = \begin{bmatrix} 1-\xi & 0 & \xi & 0 \\ 0 & 1-\xi & 0 & \xi \end{bmatrix} \begin{bmatrix} x_1 & x_2 & 0 & 0 \\ y_1 & y_2 & 0 & 0 \\ 0 & 0 & x_1 & x_3 \\ 0 & 0 & y_1 & y_3 \end{bmatrix} \begin{bmatrix} \frac{-\phi_2}{\phi_1-\phi_2} \\ \frac{\phi_1}{\phi_1-\phi_2} \\ \frac{-\phi_3}{\phi_1-\phi_3} \\ \frac{\phi_1}{\phi_1-\phi_3} \end{bmatrix}, \tag{21}$$

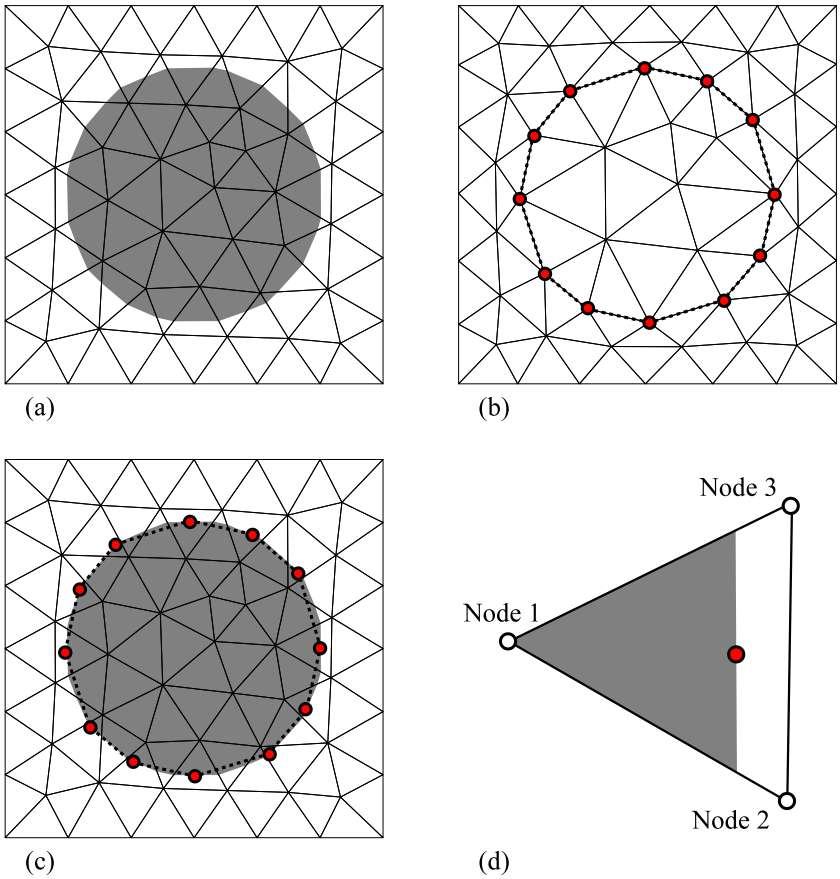

Figure 1: Relationship between the Eulerian mesh and the level-set boundary of the ALE mesh: (a) an Eulerian mesh maintaining the level-set function; (b) ALE mesh corresponding to (a); (c) superposition of the Eulerian mesh shown in (a) and the level-set boundary shown in (b); and (d) a triangular element in (c) which include a red node.

where ξ is the local coordinate on the line segment. Therefore, the sensitivities of X and Y with respect to ϕ_1, ϕ_2, and ϕ_3 are derived as

$$\begin{bmatrix} \frac{\partial X}{\partial \phi_i} \\ \frac{\partial Y}{\partial \phi_i} \end{bmatrix} = \begin{bmatrix} 1-\xi & 0 & \xi & 0 \\ 0 & 1-\xi & 0 & \xi \end{bmatrix} \begin{bmatrix} x_1 & x_2 & 0 & 0 \\ y_1 & y_2 & 0 & 0 \\ 0 & 0 & x_1 & x_3 \\ 0 & 0 & y_1 & y_3 \end{bmatrix} \begin{bmatrix} \frac{\phi_2}{(\phi_1-\phi_2)^2} & \frac{-\phi_1}{(\phi_1-\phi_2)^2} & 0 \\ \frac{-\phi_2}{(\phi_1-\phi_2)^2} & \frac{\phi_1}{(\phi_1-\phi_2)^2} & 0 \\ \frac{\phi_3}{(\phi_1-\phi_3)^2} & 0 & \frac{-\phi_1}{(\phi_1-\phi_3)^2} \\ \frac{-\phi_3}{(\phi_1-\phi_3)^2} & 0 & \frac{\phi_1}{(\phi_1-\phi_3)^2} \end{bmatrix} \begin{bmatrix} \frac{\partial \phi_1}{\partial \phi_i} \\ \frac{\partial \phi_2}{\partial \phi_i} \\ \frac{\partial \phi_3}{\partial \phi_i} \end{bmatrix}, \quad \text{for } i = 1, 2, 3. \tag{22}$$

Note that, we utilize the geometry-based re-initialization scheme proposed in study [6] to compute ξ, although the level-set function is not re-initialized in this paper.

For all red nodes, we compute the nodal coordinate sensitivities with respect to the nodal level-set function of the Eulerian mesh using Equation (22). Since the sensitivity with respect to the nodal level-set function of the Eulerian mesh is derived as follows using the chain rule

$$\frac{\partial f_i}{\partial \phi} = \frac{\partial f_i}{\partial X} \cdot \frac{\partial X}{\partial \phi} + \frac{\partial f_i}{\partial Y} \cdot \frac{\partial Y}{\partial \phi}, \quad \text{for } i = 0, 1, 2, 3, \tag{23}$$

we compute it by substituting Equations (14), (18), (19), (20), and (22) into Equation (23).

5.4. Level-set function updating scheme

In this section, we explain a level-set function updating scheme using nonlinear programming and the

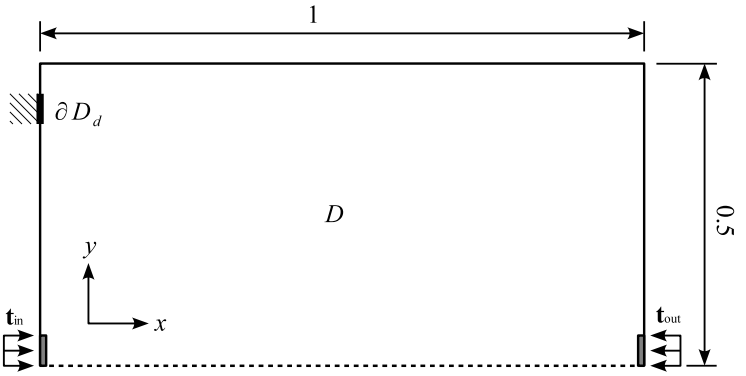

Figure 2: Design domain and boundary conditions of the numerical example in Section 6.

sensitivities derived in Section 5.3. We adopt the design variable filter proposed in papers [9, 10]. That is, we introduce the design variable ψ that governs the level-set function ϕ in the continuous system as follows:

$$-R^2\nabla^2\phi + \phi = \psi \quad \text{in } D, \tag{24}$$

where R is the length scale parameter. Here, we set the upper and lower bounds of ψ to 1 and -1, respectively. By solving Equation (24) with given ψ, smoothly distributed ϕ is obtained. Furthermore, we obtain the following relation between the sensitivities with respect to ϕ and ψ:

$$\frac{\partial f_i}{\partial \psi} = \frac{\partial f_i}{\partial \phi} \cdot \frac{\partial \phi}{\partial \psi} \quad \text{in } D, \quad \text{for } i = 0, 1, 2, 3. \tag{25}$$

Since $\frac{\partial \phi}{\partial \psi}$ corresponds to the inverse of operator $(-R^2\nabla^2 + 1)$, the same filter of Equation (24) is applied to the sensitivities with respect to ϕ, i.e., $\frac{\partial f_i}{\partial \phi}$. In this manner, the sensitives with respect to ψ, i.e., $\frac{\partial f_i}{\partial \psi}$ are obtained.

On the basis of the above discussion, the updating scheme is as follows:

(i) Provide an Eulerian mesh consisting of linear triangular elements, initialize the design variables ψ at each node of the Eulerian mesh, and set the termination counter n_{trm} to 0.

(ii) Compute the level-set function ϕ at each node of the Eulerian mesh, by solving Equation (24).

(iii) Generate the ALE mesh.

(iv) Compute the objective function $\tilde{f}_0$ and the constraint functions $\tilde{f}_1$, $\tilde{f}_2$, and $\tilde{f}_3$ using the ALE mesh.

(v) Perform the following if all constraints are satisfied.

- Set $\tilde{f}_{0,\min}$ to $\tilde{f}_0$ of the current iteration when all constraints are first satisfied.
- Compare $\tilde{f}_{0,\min}$ and $\tilde{f}_0$ of the current iteration.
- If $\tilde{f}_{0,\min}$ is smaller, increment n_{trm}. Otherwise, set $\tilde{f}_{0,\min}$ to the current $\tilde{f}_0$ and n_{trm} to 0.
- Terminate the optimization successfully, if $n_{\text{trm}} = 10$.

(vi) Compute the sensitivities with respect to ϕ, as explained in Section 5.3.

(vii) Compute the sensitivities with respect to ψ on the basis of Equation (25).

(viii) Update the design variable ψ using the sequential linear programming (SLP), and return to (ii).

6. Numerical Example

In this section, we provide a numerical example to demonstrate the usefulness of the proposed method. Figure 2 shows the design domain and boundary conditions of the numerical example. In this figure,

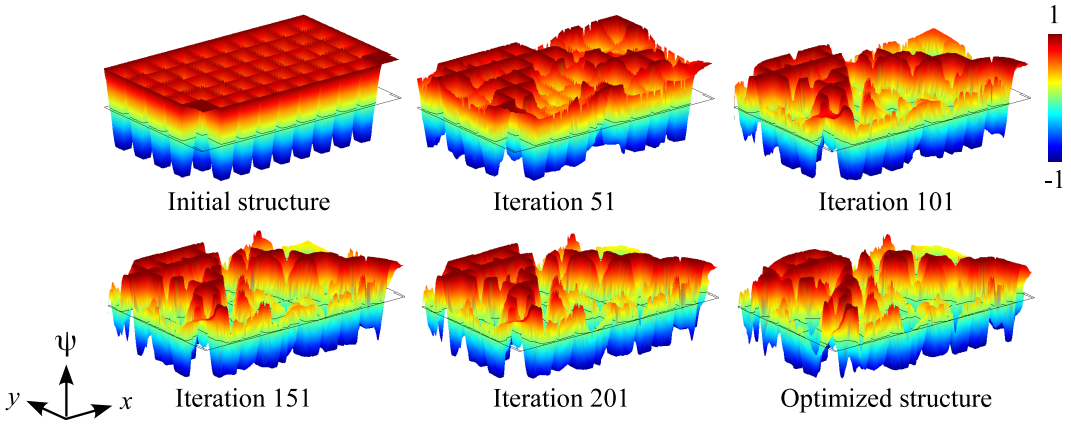

Figure 3: Design variables of initial, intermediate, and optimized structures.

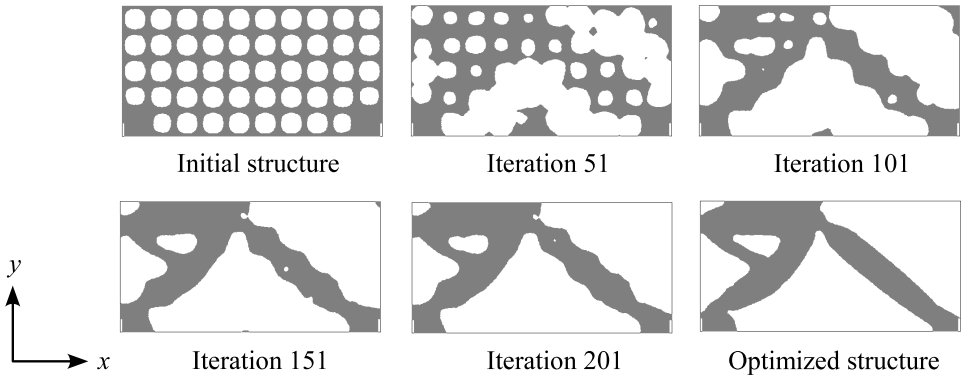

Figure 4: Material domains of initial, intermediate, and optimized structures.

∂D_{d} is the boundary where the displacement is completely fixed. The symmetry condition is imposed at the bottom dotted line. The input load $\mathbf{t}_{\mathrm{in}}$ and the output load $\mathbf{t}_{\mathrm{out}}$ are set to $(0.01, 0)$ and $(-0.01, 0)$, respectively. Young's modulus and Poisson's ratio of the material are set to 1 and 0.3, respectively. Parameter κ in Equation (8), and R in Equation (24) are set to 1×10^{-6}, and 0.025, respectively. $f_{1\mathrm{max}}$, $f_{2\mathrm{max}}$, and $f_{3\mathrm{max}}$ are set to 0.19, 2×10^{-5}, and 2×10^{-5}, respectively. The move limit of the SLP is set to 0.01. The design domain is discretized using an advancing front based method with the maximum element length 0.005.

Here, we initialize the design variables ψ as shown in the left upper corner of Figure 3. The design variables are iteratively updated as explained in Section 5.4, and the optimized structure is obtained at Iteration 345. Figure 3 shows the design variables of the initial, intermediate, and optimized structures. The corresponding material domains are obtained as shown in Figure 4. The optimization history is shown in Figure 5.

As shown in Figure 5, the objective function is minimized while all constraints are satisfied. Furthermore, the optimized structure shown in the right bottom corner of Figure 4 is very similar to a displacement inverter shown in study [7]. Therefore, we consider that an valid compliant mechanism is obtained by the proposed method.

7. Conclusion

In this paper, we proposed a level-set and ALE based topology optimization method using nonlinear programming. In the proposed method, the level-set function was updated using SLP instead of the level-set equation. Since nonlinear programming such as SLP is preferable for dealing with multiple constraints, we consider that updating the level-set function using nonlinear programming is a reasonable approach, in particular, when multiple constraints are imposed. So-called grayscale elements are completely eliminated

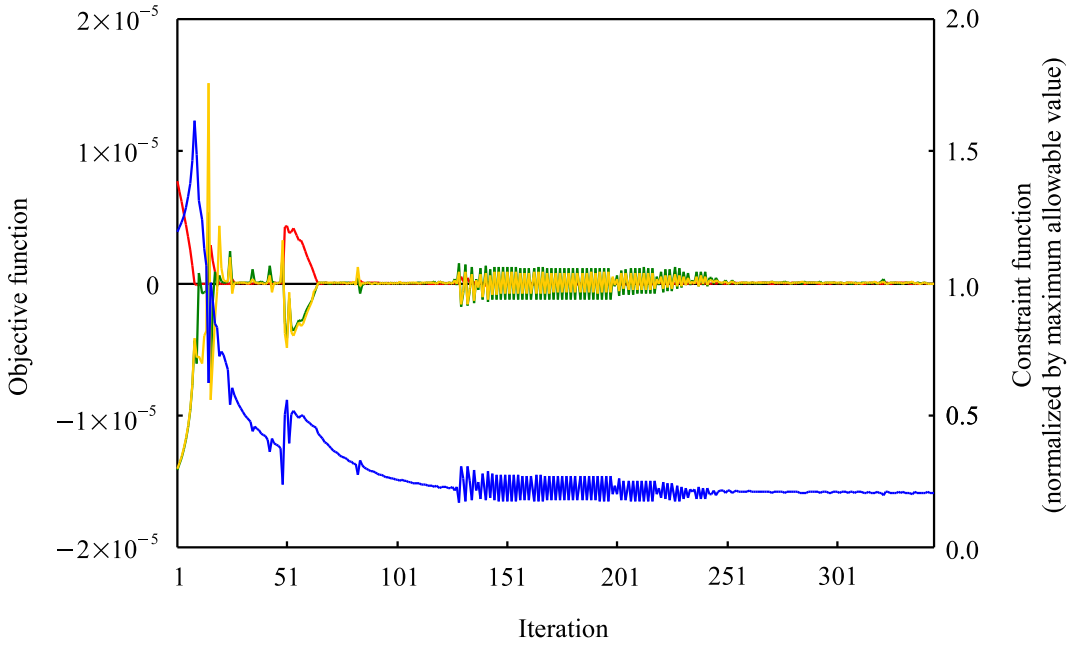

Figure 5: Optimization history of the objective function $\tilde{f}_0$ (blue) and the constraint functions $\tilde{f}_1$ (red), $\tilde{f}_2$ (green), and $\tilde{f}_3$ (orange).

in level-set and ALE based topology optimization that we previously proposed, therefore, the method proposed in this paper can be applied to various design problems where grayscale elements cause a fatal error and multiple constraints are imposed. We will apply the proposed method to such design problems in a future work.

8. Acknowledgments

This work was supported by JSPS KAKENHI Grant Number 24760065.

9. References

[1] M. P. Bendsøe and N. Kikuchi. Generating optimal topologies in structural design using a homogenization method. *Computer Methods in Applied Mechanics and Engineering*, 71(2):197–224, November 1988.

[2] M. P. Bendsøe and O. Sigmund. *Topology Optimization Theory, Methods and Applications*. Springer, Berlin, 2nd edition, 2003.

[3] M. Y. Wang, X. Wang, and D. Guo. A level set method for structural topology optimization. *Computer Methods in Applied Mechanics and Engineering*, 192(1–2):227–246, 2003.

[4] G. Allaire, F. Jouve, and A. M. Toader. Structural optimization using sensitivity analysis and a level-set method. *Journal of Computational Physics*, 194(1):363–393, February 2004.

[5] S. Osher and J. A. Sethian. Fronts propagating with curvature-dependent speed: Algorithms based on Hamilton-Jacobi formulations. *Journal of Computational Physics*, 78(1):12–49, November 1988.

[6] S. Yamasaki, T. Nomura, A. Kawamoto, K. Sato, and S. Nishiwaki. A level set-based topology optimization method targeting metallic waveguide design problems. *International Journal for Numerical Methods in Engineering*, 87(9):844–868, September 2011.

[7] S. Yamasaki, A. Kawamoto, and T. Nomura. Compliant mechanism design based on the level set and arbitrary Lagrangian Eulerian methods. *Structural and Multidisciplinary Optimization*, 46(3):343–354, September 2012.

[8] S. Osher and R. Fedkiw. *Level Set Methods and Dynamic Implicit Surfaces*. Springer, New York, 2003.

[9] B. S. Lazarov and O. Sigmund. Filters in topology optimization based on Helmholtz-type differential equations. *International Journal for Numerical Methods in Engineering*, 86(6):765–781, May 2011.

[10] A. Kawamoto, T. Matsumori, S. Yamasaki, T. Nomura, T. Kondoh, and S. Nishiwaki. Heaviside projection based topology optimization by a PDE-filtered scalar function. *Structural and Multidisciplinary Optimization*, 44(1):19–24, July 2011.