Available online at www.sciencedirect.com
ScienceDirect
Journal of Computational Physics 227 (2007) 680–705
JOURNAL OF COMPUTATIONAL PHYSICS
www.elsevier.com/locate/jcp

Shape and topology optimization of compliant mechanisms using a parameterization level set method

Zhen Luo [a,1], Liyong Tong [a,*], Michael Yu Wang [b], Shengyin Wang [c]

[a] School of Aerospace, Mechanical and Mechatronic Engineering, The University of Sydney, NSW 2006, Australia
[b] Department of Mechanical and Automation Engineering, The Chinese University of Hong Kong, Shatin N.T., Hong Kong SAR, PR China
[c] Center for Singapore-MIT Alliance, National University of Singapore, E4-04-10, 4 Engineering Drive 3, Singapore 117576, Singapore

Received 19 March 2007; received in revised form 9 August 2007; accepted 9 August 2007
Available online 30 August 2007

Abstract

In this paper, a parameterization level set method is presented to simultaneously perform shape and topology optimization of compliant mechanisms. The structural shape boundary is implicitly embedded into a higher-dimensional scalar function as its zero level set, resultantly, establishing the level set model. By applying the compactly supported radial basis function with favorable smoothness and accuracy to interpolate the level set function, the temporal and spatial Hamilton–Jacobi equation from the conventional level set method is then discretized into a series of algebraic equations. Accordingly, the original shape and topology optimization is now fully transformed into a parameterization problem, namely, size optimization with the expansion coefficients of interpolants as a limited number of design variables.

Design of compliant mechanisms is mathematically formulated as a general optimization problem with a nonconvex objective function and two additionally specified constraints. The structural shape boundary is then advanced as a process of renewing the level set function by iteratively finding the expansion coefficients of the size optimization with a sequential convex programming method. It is highlighted that the present method can not only inherit the merits of the implicit boundary representation, but also avoid some unfavorable features of the conventional discrete level set method, such as the CFL condition restriction, the re-initialization procedure and the velocity extension algorithm. Finally, an extensively investigated example is presented to demonstrate the benefits and advantages of the present method, especially, its capability of creating new holes inside the design domain.

Keywords: Shape optimization; Topology optimization; Compliant mechanisms; Level set methods; Radial basis functions; Convex programming

* Corresponding author. Tel.: +61 2 93516949; fax: +61 2 93514841.
E-mail addresses: zluo@aeromech.usyd.edu.au (Z. Luo), ltong@aeromech.usyd.edu.au (L. Tong), yuwang@mae.cuhk.edu.hk (M.Y. Wang), smaws@nus.edu.sg (S. Wang).
[1] Tel.: +61 2 93512342; fax: +61 2 93514841.

0021-9991/$ - see front matter
doi:10.1016/j.jcp.2007.08.011

1. Introduction

Compliant mechanisms [27] have been recently emerged as a relatively new family of mechanical devices that transfer or transmit motion, force, or energy from specified input ports to output ports by elastic deformation of its composing materials. However, a compliant mechanism achieves at least a portion of its mobility from structural flexibility in contrast to a rigid-link mechanism which gains motion totally via its movable joints, such as hinges, bearings and slides. It is in fact the strain energy stored in the flexible components that makes the mechanism fulfill the required function analogous to a rigid-body mechanism. The main advantage of compliant mechanisms is that fewer parts, fewer assembly processes and no lubrication are required. Due to their great promise in providing better solutions to lots of mechanical design problems, compliant mechanisms have recently experienced considerable development in a variety of areas (e.g. [29,55,25,70]). In general, compliant mechanisms can be classified into two types, partially and fully compliant mechanisms [27]. Correspondingly, the methods for designing compliant mechanisms can be approximately put into two categories.

The first approach is based on the so-called pseudo-rigid-body model (PRBM) [26,34], rooting in the concept of rigid-body kinematic synthesis. This approach, however, is limited in its capacities of yielding the elastic deformation that locates locally in small portions similar to the rigid-body mechanisms with flexural or notch hinges, leading to a partially compliant mechanism with lumped compliance. This kind of mechanism mainly distorts from a few flexural hinges so that the material surrounding the hinges is prone to suffering from overstress and overstrains that will speed up the process of fatigue breakage. Furthermore, it is difficult to extend this method to design precision devices in microsystems because of the approximating behaviors of the method itself. Although under certain specific loading configurations, the PRBM is available for modeling beams with continuous deflection [37]. By introducing the topology optimization method rooted with the homogenization theory of continuum mechanics [5], the second way is therefore established as a systemic synthesis to generate the fully compliant mechanism that has flourished in the last few years with a diversity of application [1,14,54,32]. The fully compliant mechanism is in essence a kind of jointless mechanical device, which is capable of producing distributed compliance from the elastic deformation of the entire structure due to continuity and monolithic of comprising materials [32,70,79]. It is especially suitable for designing piezoelectric transducers in smart structures [58,22] or devices in micro-electromechanical (MEMS) [24,36,56,6], as the difficulties relevant to the fabrication and assembly of joints and counterparts in micro-scale can be eliminated.

Topology optimization [10] is a process to determine the best arrangement of a given amount of material in a design domain by iteratively eliminating and redistributing them until the specified structural performance is extremized. It is well known the topology optimization is essentially a large-scale integer programming with 0 and 1 design variables [8]. Unfortunately, the optimization problems established with material distribution schemes are usually ill-posed. Hence, some element-based methods, such as the homogenization [8] or the SIMP method [9], have been popularly applied as relaxation to generate a well-posed topology optimization problem. A meaningful solution to the relaxed problem can be guaranteed further by including methods to smear out numerical instabilities [57].

Level set methods [44] have emerged recently as an attractive alternative to perform structural shape and topology optimization without relaxation, which first designed for tracking, modeling and simulating the evolution of moving boundaries with topology changes of merging and breaking naturally in many areas [52]. The level set method essentially leads to an Eulerian system of geometry partial differential equation, capable of performing topological changes and capturing geometric evolutions at the interface and the boundaries. Sethian et al. [53] can be regarded among the first researchers who introduced the level set methods [44] into structural optimization. The shape and topology changes are achieved according to the equivalent stresses on the boundaries rather than the more favorable shape derivatives. Osher and Santosa [43] studied shape and topology optimization of an inhomogeneous drum by incorporating the shape derivative into the level set method. The evolving interface is then tracked by solving a Hamilton–Jacobi equation and the descent of the objective is ensured by a steepest descent method. Wang et al. [67] presented a way by implicitly embedding the structural boundary into a scalar function of higher dimension as zero level set, and a first-order Hamilton–Jacobi PDE is then established to update the scalar function until an optimum is achieved. Wang and Wang [69] extended the previous method by establishing a ''color'' multi-phase level set model to perform multi-material

structural optimization. The highlight of their works should be the connection of the shape derivative [59] with the powerful level set method [52]. But several numerical requirements of solving the Hamilton–Jacobi PDE need to be carefully imposed to make the initial value problem meaningful [69]. Allaire et al. [2] studied the level set shape and topology optimizations by combining the shape sensitivity analysis with the front propagation technologies, including an extension to designs with different objective functions [3].

Currently, the conventional way in applying the level set method to shape and topology optimization is to directly solve the Hamilton–Jacobi PDE (e.g. [67,2]). In practice, it was noted that the implementation of the conventional level set method requires an appropriately choice of the time-marching scheme, the re-initialization procedure the extension velocity algorithm. In these numerical procedures, several complicated Hamilton–Jacobi type PDEs are usually included [52,42]. However, it is well known that these PDEs are seldom to implement [35], although some schemes like upwind schemes [33] and fast marching methods [46,52] are available. Furthermore, in the conventional level set method, no "nucleating" mechanisms were incorporated to create new holes inside the design domain, which makes the final design largely dependent on the initial guess [13,4]. Hence, the numerical requirements limit the utilities of the conventional level set method to shape and topology optimization, even if the physical meaning of numerical solutions can be guaranteed in terms of the unique entropy condition of the Hamilton–Jacobi convection equation [44].

Alternatively, Belytschko et al. [7] described the level set function in a narrow band of the zero level-set according to nodal variables of the level set function and shape functions of C0 continuity. A heuristic scheme was applied to update the nodal values of the implicit scalar function to directly propagate the design boundary. Haber [23] used a sequential quadratic programming (SQP) method in conjunction with multilevel continuation schemes to advance the implicit shape boundary rather than directly solving the Hamilton–Jacobi PDE, but it was noted that some modifications should be included to make the numerical process efficient [75]. Wang and Wang [73,74] introduced global shape functions into the conventional level set method and developed a kind of parameterization scheme. In doing so, the original initial value problem is converted into a system of coupled ordinary differential equations (ODEs). A steepest descent method was utilized to ensure the decrease of the objective function, which is a classic gradient method only allowing fast descent locally rather than a global steepest descent [38]. The interpolation using the global shape functions led to a full dense collection matrix which would limit its application to large-scale systems. The velocity field was explicitly worked out in terms of the strain energy density and the Lagrange multiplier of the constraint, and a heuristic scheme was then applied to extend the velocity field from the design boundary to the entire design domain. Furthermore, only the simple unconstrained optimization was implemented in the framework of compliance designs. It is difficult to extend their method to more advanced optimization with multiple constraints, due to the difficulty in determining active constraints. Hence, it is necessary to put a through investigation to explore more attractive potential in applying level set methods and radial basis functions to shape and topology optimization. The aim of this paper is therefore to present a new parameterization level set method for shape and topology optimization of compliant mechanisms using the compactly supported radial basis function (CSRBF).

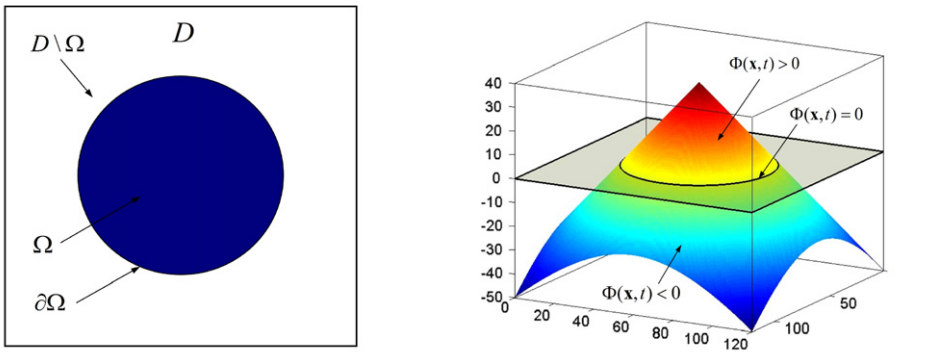
2. Implicit boundary representation

In implicit boundary representations of the level set method, structural design boundary is first embedded implicitly as zero level set of a higher-dimensional level set function, and then an appropriate speed field is incorporated into the implicitly expressed boundary to iteratively advance the interface on a fixed Eulerian rectilinear grids [41,52].

A Lipschitz continuous level set function $\Phi(\mathbf{x})$ is defined over a reference domain $D \subset R^d$ ($d = 2$ or 3) including all admissible shapes Ω ($\Omega \subset D$). The 3D structure is then embedded as follows:

$$\begin{cases} \Phi(\mathbf{x}) > 0 & \forall \mathbf{x} \in \Omega \setminus \partial\Omega \quad \text{(inside)} \\ \Phi(\mathbf{x}) = 0 & \forall \mathbf{x} \in \partial\Omega \cap D \quad \text{(boundary)} \\ \Phi(\mathbf{x}) < 0 & \forall \mathbf{x} \in D \setminus \Omega \quad \text{(outside)} \end{cases} \tag{1}$$

The representation scheme is illustrated in Fig. 1a and b for a case of $d = 2$.



(a) Solid Ω and its embedding domain D (b) Level set function and its zero level set

Fig. 1. Design domain and the level set model.

By introducing the pseudo-time t for the dynamic process of shape deformations, one obtains the following Hamilton–Jacobi equation [52,42]

$$\frac{\partial \Phi(\mathbf{x},t)}{\partial t} + v_n|\nabla\Phi| = 0, \Phi(\mathbf{x},0) = \Phi_0(\mathbf{x}) \tag{2}$$

as a result of the normal velocity $v_n = v \cdot (\nabla\Phi/|\nabla\Phi|)$, where $(v = \mathrm{d}\mathbf{x}/\mathrm{d}t)$, $\mathbf{n} = \nabla\Phi/|\nabla\Phi|$, and $|\nabla\Phi| = \sqrt{\nabla\Phi \cdot \nabla\Phi}$. The above equation is also known as the level set-based implicit boundary representation. Moving the structural boundary $\Gamma = \{\mathbf{x}|\Phi(\mathbf{x}) = 0\}$ along the normal direction is then equivalent to transporting $\Phi(\mathbf{x},t)$ by finding solutions of the Hamilton–Jacobi equation, directly or indirectly.

In the conventional level set method [67,2], the partial differential equation (PDE) in Eq. (2) is directly solved by using explicit time-marching schemes, such as the finite difference approach [33]. The evolvement of the level set function is thus governed by a series of solutions of the Hamilton–Jacobi PDE [52]. In general, the velocity in the Hamilton–Jacobi equation which advances the level set function is obtained from the sensitivity analysis of the objective function, and the descent direction of the objective is determined by the steepest descent method [67].

In general, the implicit moving boundary of the level set method possesses several favorite features, which can be summarized as [67,68]:

(1) The implicit level set methods can simultaneously address shape fidelity and topology changes, especially, keep the design boundary smooth during the whole optimization process.
(2) The higher-dimensional level set function usually has a simple topology but it cannot prevent its capability of representing very complicated boundary changes.
(3) The interface represented by the level set function is parametric free because the normal component of a general velocity vector only exerts influence on the shape geometry while the tangential influences the shape parameterization.
(4) The theory of viscosity solutions [18] can be applied to guarantee a physically meaningful result for the level set model of the Hamilton–Jacobi equation.

However, in applying the conventional discrete level set method to shape and topology optimization [67,2], a general analytical function for the level set function is usually unknown, and thus it needs to be discretized for enabling the level set process through a distance transform. Thus, it is indispensable to employ numerical procedures to solve the Hamilton–Jacobi PDE. Hence, the upwind scheme, the re-initialization procedure and the velocity extension algorithm [52,42] should be carefully handled to make the numerical procedures meaningful. Several numerical issues of the conventional level set method can be further summarized as

(1) The level set function for the same design boundary cannot be uniquely decided since any scalar function with Lipschitz continuity is available as long as it can keep the zero level set unchanged. When processing the boundary evolvement using the finite difference scheme, the scalar function is prone to drifting

away from its initial shape of a signed distance function in under-resolved regions [28] or inducing the unwanted dissipation of the front [61], leading to either a too flat or a too steep surface or hyper-surface. To stabilize numerical process, the level set function is usually re-initialized as a signed distance function to avoid a too flat or too steep surface or hyper-surface [46,66]. An easier way is to periodically develop a new Hamilton–Jacobi PDE to re-initialize the level set function to compensate the numerical errors [52]. However, it was proved that the global re-initializations are time-consuming and also disable creating new holes inside the material domain. Hence, the periodical re-initializations for the level set function should be avoided as much as possible.

(2) The mesh should be fine enough to smear numerical truncation errors caused by the polynomial snaking phenomena if low-order shape functions are used [7,15]. In practice, with an explicit time-marching scheme, the numerical solution can be theoretically close to the real one only if time and space steps are infinitely small and all numerical errors are ignored. However, in terms of the Courant–Friedrichs–Lewy (CFL), the time-step length is decided by the smallest gird size [52]. Hence, a large amount of iterations is often indispensable to advance the interface on a finer Eulerian grid. Too much iteration, in turn, will seriously deteriorate the numerical stability due to the accumulation of the truncation errors. Thus, the CFL condition should be greatly relaxed or eliminated.

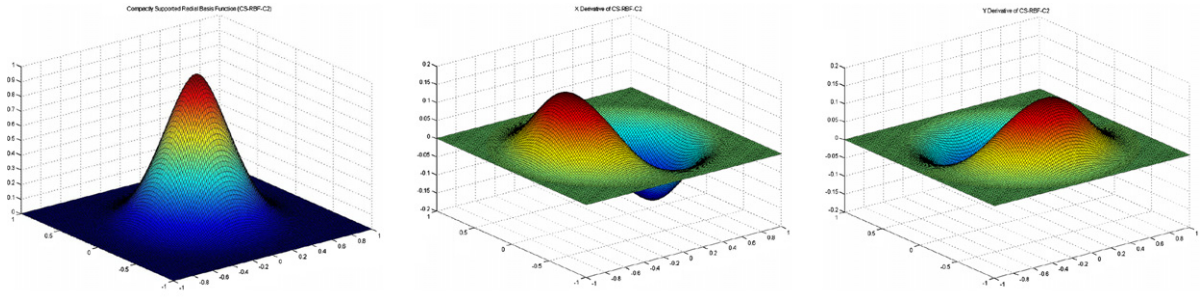
(3) The velocity field, originally derived from the shape derivative, is only meaningful at the design boundary. In the conventional level set method, however, the velocity field needs to be explicitly evaluated in a neighborhood of the design boundary [46,2]. Hence, it is often required to extend the original velocity field to a set of grids. A most natural way is to extend the velocity field by letting the velocity be constant along the normal direction, thus leading to another new Hamilton–Jacobi PDE [52].

In this work, the design boundary is still represented as a level set model by fully taking into account the advantage of the implicit moving boundary. However, the design boundary is advanced by using a parameterization scheme rather than solving the Hamilton–Jacobi PDE directly as the conventional level set method [67,2]. Thus, the related numerical limitations, such as the CFL condition, the re-initialization process and the velocity extension algorithm, are all expected to be eliminated.

3. Parameterization of the level set function

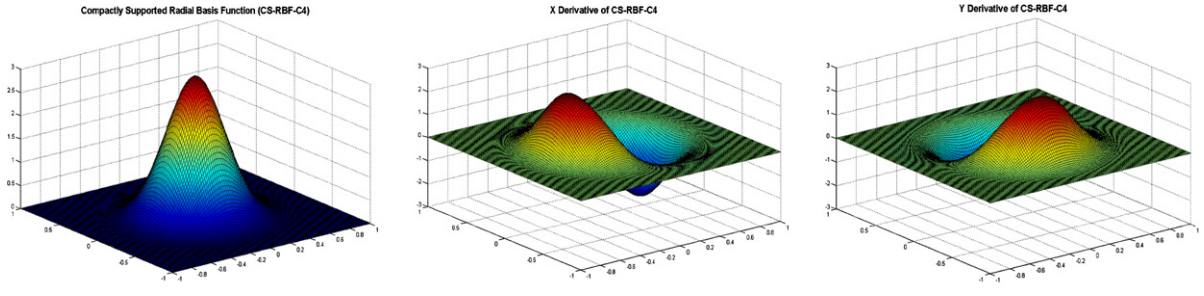
The radial basis functions [12], radial-symmetrically centered at a particular point or knot, such as globally defined radial basis functions or compactly supported radial basis functions, have been popularly employed in approximating multivariate scattered data in recent years. The globally defined radial basis functions have been used to interpolate the level set function for structural shape and topology optimization [72,73], but the free shape parameter has a significant influence on the accuracy of the interpolation and there is still absent a well-established scheme to determine its optimal value [47]. Furthermore, the fully dense matrix will obviously burden the computational efforts for a system with a large amount of candidate points. Here, the compactly supported radial basis functions (CSRBFs) [76,49] are employed as an alternative way to overcome the shortcomings of the global basis functions. CSRBFs have been popular in multivariate interpolations due to its strictly positive definiteness and sparseness [12]. The positive definiteness and continuity can be well incarnated by finding a nonnegative and nonvanishing Fourier transform in terms of the theorem of Bochner [51]. The particular attractiveness of the CSRBFs is that the interplant can naturally inherit the continuity of the radial basis functions, which enables us to select the appropriate shape functions from a set of well known CSRBFs [77,50].

A variety of CSRBFs can be available [12] in approximating complicated functions. In this paper, a popularly studied family of CSRBFs with 2k continuity [76,77], such as C2, C4 and C6 functions, is adopted to interpolate the level set function. As indicated by their shapes and derivatives in Figs. 2–4, any of these CSRBFs can be applied to interpolate the scalar function with desired smoothness and completeness if CSRBF knots are properly arranged in the design domain. It is obvious that the C6 function has the highest differentiability relevant to the steepest caps of shape and derivatives. Hence, it might be more sensitive to the variation of the support radius. In general, bigger supports should be applied to low-order CSRBFs and smaller supports to high-order CSRBFs [78]. The experiential criterion of choosing an appropriate support radius is to make a trade-off in both ensuring the non-singularity of the interpolation and guaranteeing a modest



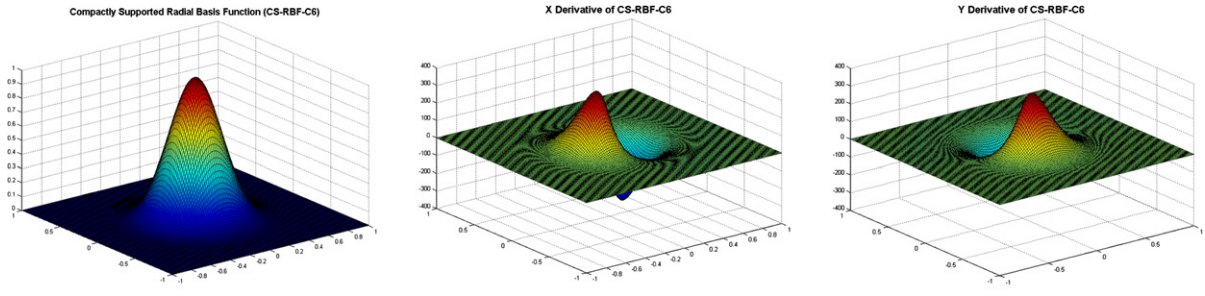
(a) Shape (b) Derivative in X direction (c) Derivative in Y direction

Fig. 2. CSRBF with C2 continuity and its derivatives in X and Y directions.



(a) Shape (b) Derivative in X direction (c) Derivative in Y direction

Fig. 3. Shape of CSRBF-C4 and its derivatives in X and Y directions.



(a) Shape (b) Derivative in X direction (c) Derivative in Y direction

Fig. 4. Shape of CSRBF-C6 and its derivatives in X and Y directions.

numerical effort. A too small support radius cannot enable CSRBFs to effectively span the inner-constraint gaps, and a too large radius would remarkably increase the computation time.

3.1. Wendland-C2

(1) Shape and derivatives of the CSRBF with C2 continuity are given, respectively, as

$$\phi(r) = \max\{0,(1-r)\}^4(4r+1) \tag{3}$$

$$\frac{\partial\phi}{\partial x} = \frac{\partial\phi}{\partial r}\frac{\partial r}{\partial x} = \max\{0,(1-r)\}^3(-20r)\frac{\partial r}{\partial x} \tag{4}$$

$$\frac{\partial\phi}{\partial y} = \frac{\partial\phi}{\partial r}\frac{\partial r}{\partial y} = \max\{0,(1-r)\}^3(-20r)\frac{\partial r}{\partial y} \tag{5}$$

3.2. Wendland-C4

(2) Shape and derivatives of the CSRBF with C4 continuity are given, respectively, as

$$\phi(r) = \max\{0,(1-r)\}^6(35r^2+18r+3) \tag{6}$$

$$\frac{\partial\phi}{\partial x} = \frac{\partial\phi}{\partial r}\frac{\partial r}{\partial x} = \max\{0,(1-r)\}^5(-280r^2-56r)\frac{\partial r}{\partial x} \tag{7}$$

$$\frac{\partial\phi}{\partial y} = \frac{\partial\phi}{\partial r}\frac{\partial r}{\partial y} = \max\{0,(1-r)\}^5(-280r^2-56r)\frac{\partial r}{\partial y} \tag{8}$$

3.3. Wendland-C6

(3) Shape and derivatives of the CSRBF with C6 continuity are given, respectively, as

$$\phi(r) = \max\{0,(1-r)\}^8(32r^3+25r^2+8r+1) \tag{9}$$

$$\frac{\partial\phi}{\partial x} = \frac{\partial\phi}{\partial r}\frac{\partial r}{\partial x} = \max\{0,(1-r)\}^7(-352r^3-154r^2-22r)\frac{\partial r}{\partial x} \tag{10}$$

$$\frac{\partial\phi}{\partial y} = \frac{\partial\phi}{\partial r}\frac{\partial r}{\partial y} = \max\{0,(1-r)\}^7(-352r^3-154r^2-22r)\frac{\partial r}{\partial y} \tag{11}$$

In above CSRBFs, the radius of support is given in a 2-D Euclidean space

$$r = \frac{d_I}{d_{mI}} = \frac{\sqrt{(x-x_i)^2+(y-y_i)^2}}{d_{mI}} \tag{12}$$

where d_I can be regarded as a function which measures the distance of the current sample knot (x,y) to knot (x_i,y_i), and the derivatives of r in different directions is defined as

$$\frac{\partial r}{\partial x} = \frac{1}{d_{mI}}\frac{(x-x_i)}{\sqrt{(x-x_i)^2+(y-y_i)^2}} \tag{13}$$

$$\frac{\partial r}{\partial y} = \frac{1}{d_{mI}}\frac{(y-y_i)}{\sqrt{(x-x_i)^2+(y-y_i)^2}} \tag{14}$$

The support size at a specified knot is calculated by $d_{mI} = d_{\max}\cdot C_I$, where $d_{\max}$ is a scaling parameter factor, typically 2.0–4.0 for a static analysis, and C_I is the distance which is used to guarantee a meaningful interpolation by searching for enough nodes in the neighborhood of the current knot [19].

The level set function can thus be described by centrally positioning the shape functions at their pre-specified knots over the whole design domain, which means

$$\Phi(\mathbf{x}) = \boldsymbol{\varphi}(\mathbf{x})^{\mathrm{T}}\boldsymbol{\alpha} = \sum_{i=1}^{N}\phi_i(\mathbf{x})\alpha_i \tag{15}$$

with a vector of the shape functions

$$\boldsymbol{\varphi}(\mathbf{x}) = [\phi_1(\mathbf{x}),\phi_2(\mathbf{x}),\ldots,\phi_N(\mathbf{x})]^{\mathrm{T}} \in \mathbb{R}^N \tag{16}$$

and the expansion coefficient vector

$$\boldsymbol{\alpha} = [\alpha_1,\alpha_2,\ldots,\alpha_N]^{\mathrm{T}} \in \mathbb{R}^N \tag{17}$$

The interpolant is uniquely determined in terms of the given interpolating data of the level set function located at knots, owning to the property of strictly positive definiteness. Since the present interpolating scheme is performed under the assumption that all the knots are fixed in the design domain, the space and time of

the original Hamilton–Jacobi PDE is now thoroughly separated. In this way, the radial basis functions are spatial functions only and the expansion coefficients are temporal only.

$$\Phi(\mathbf{x},t) = \boldsymbol{\varphi}(\mathbf{x})^{\mathrm{T}}\boldsymbol{\alpha}(t) \tag{18}$$

the dynamic level set model is given as

$$\boldsymbol{\varphi}(\mathbf{x})^{\mathrm{T}}\dot{\boldsymbol{\alpha}}(t) - \mathbf{v}_n|(\nabla\boldsymbol{\varphi})^{\mathrm{T}}\boldsymbol{\alpha}(t)| = 0 \tag{19}$$

The normal velocity field related to the time-derivative of the coefficients is now given as

$$\mathbf{v}_n = \frac{\boldsymbol{\varphi}(\mathbf{x})^{\mathrm{T}}}{|(\nabla\boldsymbol{\varphi})^{\mathrm{T}}\boldsymbol{\alpha}(t)|}\dot{\boldsymbol{\alpha}}(t), \quad \dot{\boldsymbol{\alpha}}(t) = \frac{\mathrm{d}\boldsymbol{\alpha}(t)}{\mathrm{d}t} \tag{20}$$

Hereto, the compactly supported radial basis function (CSRBF) is applied to fully parameterize the original Hamilton–Jacobi PDE in Eq. (2) into an interpolation problem. One scheme is to advance in time the expansion coefficients utilizing Eq. (19) as proposed by Wang and Wang [73]. With an additional heuristic scheme to extend the velocity field, Eq. (19) can then be applied to each of RBF knots over the entire design domain.

In this paper, as a physically meaningful extension, Eq. (19) is naturally applied to each of these knots of the CSRBF interpolation. The normal velocity v_n in Eq. (20), which does not need to be explicitly worked out, is now naturally extended to all knots in the entire design domain, which makes it possible to create new holes inside the design domain.

4. Optimal synthesis of compliant mechanisms

In this section, the shape and topology optimization of compliant mechanisms is established as a general nonlinear programming problem. The shape derivative is used to perform the design sensitivity analysis. For the sake of simplicity, the linear elastic structure is used with a single input and a single output but without losing any generality, and an extension to nonlinear structures [45,11] is straightforward.

4.1. Mathematical formulation of the optimization problem

A variety of research works are concerned with defining an appropriate objective function to embody the desired structural performance [21,32]. It has been reported that functional specifications of the mechanism, such as the mechanical advantage (MA), the geometrical advantage (GA) and the mechanical efficiency (ME), can be used to quantify the mechanical performance [54,30,70]. In particular, ME represents a well-balanced combination of GA and MA, which is capable of controlling the trade-offs between the desired displacement and force output. In other words, the MA is used to guarantee the mechanical stiffness to resist the reaction force, and GA measures the structural flexibility for producing desired deformation. As illustrated in Fig. 5, an artificial spring model with known-stiffness [54] is introduced to simulate the mutual reaction between a workpiece and the mechanism.

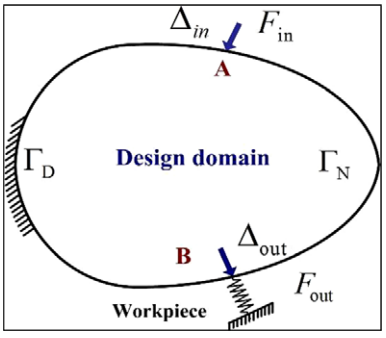


Fig. 5. Design domain with spring model.

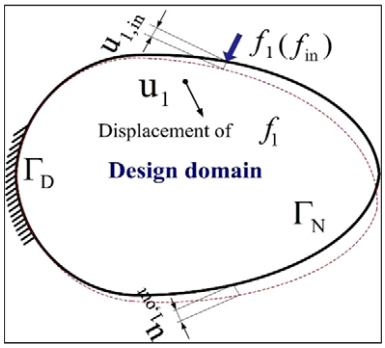


Fig. 6. Input unit dummy load.

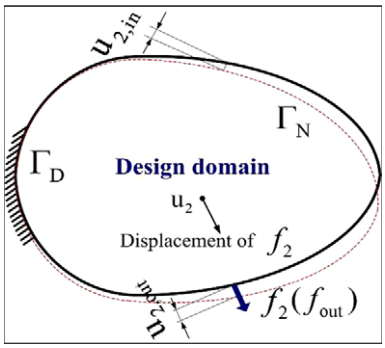


Fig. 7. Output unit dummy load.

The dummy load method [54] is used to describe the displacements engendered by two dummy loading cases acting on the input (Fig. 6) and the output (Fig. 7) ports, respectively. u_1 indicates the displacement field caused by the unit dummy f_1 at the input port, and the displacements at the input and the output ports are denoted as $u_{1,\text{in}}$ and $u_{1,\text{out}}$ respectively. Similarly, u_2 is the structural displacement field induced by only applying the unit dummy load f_2 at the output port, where $u_{2,\text{in}}$ and $u_{2,\text{out}}$ are the displacements located at the input and the output ports.

The displacements Δ_{in} and Δ_{out} involved at the input and the output ports can be formulated respectively by superposition of the dummy loading cases

$$\Delta_{\text{in}} = \mathrm{F}_{\text{in}} u_{1,\text{in}} + \mathrm{F}_{\text{out}} u_{2,\text{in}} \tag{21}$$

$$\Delta_{\text{out}} = \mathrm{F}_{\text{in}} u_{1,\text{out}} + \mathrm{F}_{\text{out}} u_{2,\text{out}} \tag{22}$$

where the displacements $u_{1,\text{in}}$, $u_{1,\text{out}}$, $u_{2,\text{in}}$ and $u_{2,\text{out}}$ can be derived from the displacement fields u_1 and u_2. It is noted that u_1 and u_2 can be obtained respectively by solving the following two self-adjoint weak forms [70]

$$\int_D E_{ijkl}\varepsilon_{ij}(u_1)\varepsilon_{kl}(v_1)H(\Phi)\mathrm{d}\Omega = f_1 v_1, \quad u|_{\Gamma_D} = u_0 \ \forall v_1 \in \mathbf{V}_1 \tag{23}$$

$$\int_D E_{ijkl}\varepsilon_{ij}(u_2)\varepsilon_{kl}(v_2)H(\Phi)\mathrm{d}\Omega = f_2 v_2, \quad u|_{\Gamma_D} = u_0 \ \forall v_2 \in \mathbf{V}_2 \tag{24}$$

where, E_{ijkl} is the elasticity tensor, ε_{ij} is the strain tensor, v_1 and v_2 denote the virtual displacement fields in spaces$\mathbf{V}_1$ and $\mathbf{V}_2$ spanned by all kinematically admissible displacements. $H(\Phi)$ is the Heaviside function which in fact serves as a characteristic function to uniformly indicates different parts in the reference domain D that has been introduced as a bounded open set to include all admissible shapes [67,2].

Then, $u_{1,\text{in}}$, $u_{1,\text{out}}$, $u_{2,\text{in}}$ and $u_{2,\text{out}}$ are found in terms of u_1 and u_2, sequentially, as

$$u_{1,\text{in}} = \int_D E_{ijkl}\varepsilon_{ij}(u_1)\varepsilon_{kl}(u_1)H(\Phi)\mathrm{d}\Omega \tag{25}$$

$$u_{1,\text{out}} = \int_D E_{ijkl}\varepsilon_{ij}(u_1)\varepsilon_{kl}(u_2)H(\Phi)\mathrm{d}\Omega \tag{26}$$

$$u_{2,\text{in}} = \int_D E_{ijkl}\varepsilon_{ij}(u_2)\varepsilon_{kl}(u_1)H(\Phi)\mathrm{d}\Omega \tag{27}$$

$$u_{2,\text{out}} = \int_D E_{ijkl}\varepsilon_{ij}(u_2)\varepsilon_{kl}(u_2)H(\Phi)\mathrm{d}\Omega \tag{28}$$

GA is measured by the ratio of displacements Δ_{out} and Δ_{in} at the output and input ports, and MA is the ratio of reaction force F_{out} and F_{in} at the output and input ports. The force $F_{\text{out}} = k_{\text{s}} \cdot \Delta_{\text{out}}$ at the output port can be achieved in terms of the spring model, where k_{s} is the stiffness of the artificial spring. GA and MA are defined, respectively, as

$$\text{GA} = \frac{\Delta_{\text{out}}}{\Delta_{\text{in}}} = \left(\frac{u_{1,\text{out}}}{u_{1,\text{in}} - k_{\text{s}}u_{1,\text{in}}u_{2,\text{out}} + k_{\text{s}}u_{1,\text{out}}u_{2,\text{in}}}\right) \tag{29}$$

$$\text{MA} = \frac{F_{\text{out}}}{F_{\text{in}}} = \frac{k_{\text{s}}u_{1,\text{out}}}{1 - k_{\text{s}}u_{2,\text{out}}} \tag{30}$$

ME is then measured by the ratio of output work to input work which is given as

$$\text{ME} = \text{sign}(\text{GA})(\text{MA} \times \text{GA}) \tag{31}$$

where the sign(GA) is used to indicate the desired direction of the output displacement.

With the present level set approach, the optimization problem is developed as

$$\begin{cases} \underset{(u_1,u_2)}{\text{Minimize}}: & J_\Phi(u_1,u_2) = -ME_\Phi(u_1,u_2) \\ \text{Subject to}: & \begin{cases} G'_\Phi(u_1,u_2) = \int_D g_1(u_1,u_2)H(\Phi)\mathrm{d}\Omega \leqslant 0 \\ G''_\Phi(u_1,u_2) = \int_D g_2(u_1,u_2)H(\Phi)\mathrm{d}\Omega \leqslant 0 \\ a_\Phi(u_1,v_1) = l_\Phi(v_1) \quad \forall v_1 \in \mathbf{V}_1 \\ a_\Phi(u_2,v_2) = l_\Phi(v_2) \quad \forall v_2 \in \mathbf{V}_2 \end{cases} \end{cases} \tag{32}$$

where G'_Φ is introduced to limit the input displacement to control the maximum stress level in the resulting mechanism [54], and G''_Φ is a volume fraction constraint used to guarantee the topology change.

The bilinear functionals with respect to the two loading cases can be given respectively as

$$a_\Phi(u_1,v_1) = \int_D E_{ijkl}\varepsilon_{ij}(u_1)\varepsilon_{kl}(v_1)H(\Phi)\mathrm{d}\Omega \tag{33}$$

$$a_\Phi(u_2,v_2) = \int_D E_{ijkl}\varepsilon_{ij}(u_2)\varepsilon_{kl}(v_2)H(\Phi)\mathrm{d}\Omega \tag{34}$$

The loading functionals are respectively specified as

$$l_\Phi(v_1) = \int_D pv_1H(\Phi)\mathrm{d}\Omega + \int_D \tau v_1\delta(\Phi)|\nabla\Phi|\mathrm{d}\Omega \tag{35}$$

$$l_\Phi(v_2) = \int_D pv_2H(\Phi)\mathrm{d}\Omega + \int_D \tau v_2\delta(\Phi)|\nabla\Phi|\mathrm{d}\Omega \tag{36}$$

where p and τ represent the body forces and boundary tractions, respectively.

4.2. Design sensitivity analysis using shape derivative

The goal of this section is to use design sensitivity analysis way to obtain the first-order derivatives of design functionals. Amongst many available methods, this section applies the concept of shape derivative [60] to per-

form the design sensitivity analysis. As far as the development of the shape derivative is concerned, one way is to use the Fréchet derivative following the way of shape diffeomorphism [67,2]. As an alternation, the material derivative of continuum mechanics [16] can also be utilized to find the shape derivative, as indicated in Wang and Wang [68]. Here, the Fréchet derivative is applied to find the shape derivative by considering not only the change of shapes upon state variables but also the perturbation of the design domain.

Supposing the general functional is defined as follows:

$$J_{\Phi}(u) = \int_D \vartheta(u) H(\Phi) \mathrm{d}\Omega \tag{37}$$

and the corresponding state equilibrium equation is expressed as

$$a_{\Phi}(u, \upsilon) = l_{\Phi}(\upsilon), \quad \forall \upsilon \in \mathbf{V}, \quad u|_{\Gamma_D} = u_0 \tag{38}$$

The shape derivative for a general functional can be worked out by following the popular scheme of Belytschko et al., Allaire et al. [67,2]. Using of the concept of the Fréchet derivative, the details for developing the shape derivative of $J_{\Phi}(u)$ is provided in Appendix A. Here, we only directly present the final outcome

$$\frac{\mathrm{d}J_{\Phi}(u)}{\mathrm{d}t} = \int_D \mathrm{v}_n \rho_{\Phi}(u, w) \delta(\Phi) |\nabla \Phi| \mathrm{d}\Omega \tag{39}$$

with the definition of the shape gradient density $\rho_{\Phi}(u, w)$ as

$$\rho_{\Phi}(u, w) = \vartheta(u) + pw - E_{ijkl}\varepsilon_{ij}(u)\varepsilon_{kl}(w) + \left(\nabla(\tau w) \cdot \frac{\nabla \Phi}{|\nabla \Phi|} + \left(\nabla \cdot \frac{\nabla \Phi}{|\nabla \Phi|}\right)\tau w\right) \tag{40}$$

Recall $\mathrm{v}_n = \boldsymbol{\varphi}(\mathbf{x})^{\mathrm{T}} \dot{\boldsymbol{\alpha}}(t) / |(\nabla \boldsymbol{\varphi})^{\mathrm{T}} \boldsymbol{\alpha}(t)|$ expressed in Eq. (20), the shape derivative of $J_{\Phi}(u)$ in Eq. (39) can then be rewritten by

$$\frac{\mathrm{d}J(u, \Phi)}{\mathrm{d}t} = \int_{\partial D} \rho_{\Phi}(u, w) \frac{\boldsymbol{\varphi}(\mathbf{x})^{\mathrm{T}}}{|(\nabla \boldsymbol{\varphi})^{\mathrm{T}} \boldsymbol{\alpha}|} \dot{\boldsymbol{\alpha}}(t) \mathrm{d}\Gamma \tag{41}$$

The above mentioned Equation is expressed as boundary integration. Considering the relation $\mathrm{d}\Gamma = \delta(\Phi)|\nabla \Phi| \mathrm{d}\Omega$ [68] between boundary ∂D and volume D, we can get an alternative scheme of volume integration with the following form

$$\frac{\mathrm{d}J(u, \Phi)}{\mathrm{d}t} = \int_D \rho_{\Phi}(u, w) \frac{\boldsymbol{\varphi}(\mathbf{x})^{\mathrm{T}}}{|(\nabla \boldsymbol{\varphi})^{\mathrm{T}} \boldsymbol{\alpha}|} \delta(\Phi) \dot{\boldsymbol{\alpha}}(t) |\nabla \Phi| \mathrm{d}\Omega = \int_D \rho_{\Phi}(u, w) \boldsymbol{\varphi}(\mathbf{x})^{\mathrm{T}} \delta(\Phi) \dot{\boldsymbol{\alpha}}(t) \mathrm{d}\Omega \tag{42}$$

where $\delta(\Phi)$ is the Dirac function. As Eq. (13) shows, $|\nabla \Phi| = |(\nabla \boldsymbol{\varphi})^{\mathrm{T}}\, \boldsymbol{\alpha}|$.

As a matter of fact, we can employ either of them to implement the numerical procedure with the boundary or the volume integral scheme, respectively. However, it was noted that the boundary integration involves a time-consuming [71] procedure to calculate $|\nabla \Phi|$ using Guass quadrature during each iteration, Instead, the volume integral scheme has been substantiated with a higher efficiency [71]. Therefore, in this work, the volume integral scheme is used.

In terms of Eq. (42), the shape derivative of ME is written with a summation form as

$$\frac{\mathrm{dME}_{\Phi}(u, w)}{\mathrm{d}t} = \sum_{i=1}^{N} \dot{\alpha}_i(t) \int_D \xi_{\Phi}(u, w) \phi_i \delta(\Phi) \mathrm{d}\Omega \tag{43}$$

Alternatively, the shape derivative of the objective is also expressed using the chain rule as

$$\frac{\mathrm{dME}_{\Phi}(u, w)}{\mathrm{d}t} = \frac{\partial \mathrm{ME}_{\Phi}(u, w)}{\partial \alpha} \frac{\mathrm{d}\alpha}{\mathrm{d}t} = \sum_{i=1}^{N} \dot{\alpha}_i(t) \left(\frac{\partial \mathrm{ME}_{\Phi}(u, w)}{\partial \alpha_i(t)}\right) \tag{44}$$

where $i = 1, 2, \ldots, N$ indicate the number of CSRBF knots.

As a result, comparing the corresponding terms in Eqs. (43) and (44), we can obtain the design sensitivity of ME as follows:

$$\frac{\partial \mathrm{ME}_{\Phi}(u,w)}{\partial \alpha_i} = \int_D \xi_{\Phi}(u,w)\phi_i \delta(\Phi)\mathrm{d}\Omega \tag{45}$$

by defining the shape gradient density $\xi_{\Phi}(u,w)$ of ME as

$$\begin{aligned}\xi_{\Phi}(u,w) = &\{[\mathrm{MA}_{\Phi}(u,w)A]E_{ijkl}\varepsilon_{ij}(u_1)\varepsilon_{kl}(u_1) + [\mathrm{MA}_{\Phi}(u,w)(B+D) + \mathrm{GA}_{\Phi}(u,w)E]E_{ijkl}\varepsilon_{ij}(u_1)\varepsilon_{kl}(u_2)\\ &+[\mathrm{MA}_{\Phi}(u,w)C + GA_{\Phi}(u,w)F]E_{ijkl}\varepsilon_{ij}(u_2)\varepsilon_{kl}(u_2)\}\end{aligned} \tag{46}$$

where GA_{Φ} and MA_{Φ} can be obtained by Eqs. (29) and (30), and the six coefficients are, respectively, defined as

$$A = (u_{1,\mathrm{out}} - k_s u_{1,\mathrm{out}}u_{2,\mathrm{out}})/(u_{1,\mathrm{in}} - k_s u_{1,\mathrm{in}}u_{2,\mathrm{out}} + k_s u_{1,\mathrm{out}}^2) \tag{47}$$

$$B = k_s u_{1,\mathrm{out}}^2/(u_{1,\mathrm{in}} - k_s u_{1,\mathrm{in}}u_{2,\mathrm{out}} + k_s u_{1,\mathrm{out}}^2) \tag{48}$$

$$C = -k_s u_{1,\mathrm{out}}u_{1,\mathrm{out}}/(u_{1,\mathrm{in}} - k_s u_{1,\mathrm{in}}u_{2,\mathrm{out}} + k_s u_{1,\mathrm{out}}^2) \tag{49}$$

$$D = k_s u_{1,\mathrm{in}}u_{1,\mathrm{out}}/(u_{1,\mathrm{in}} - k_s u_{1,\mathrm{in}}u_{2,\mathrm{out}} + k_s u_{1,\mathrm{out}}^2) \tag{50}$$

$$E = k_s/(1 - k_s u_{2,\mathrm{out}}) \tag{51}$$

$$F = k_s^2 u_{1,\mathrm{out}}/(1 - k_s u_{2,\mathrm{out}})^2 \tag{52}$$

Similarly, the sensitivity for the volume constraint G'_{Φ} and the maximal input displacement constraint G''_{Φ} can be worked out, straightforwardly, as follows:

$$\frac{\mathrm{d}G'_{\Phi}(u,w)}{d\alpha_i} = \int_D \zeta_{\Phi}(u,w)\phi_i\delta(\Phi)\mathrm{d}\Omega \tag{53}$$

$$\frac{\mathrm{d}G''_{\Phi}(u,w)}{\partial\alpha_i} = \int_D \eta_{\Phi}(u,w)\phi_i\delta(\Phi)\mathrm{d}\Omega \tag{54}$$

where $\zeta_{\Phi}(u,w) = 1$ is the shape density function of the volume constraint, and η is the shape density function of the input displacement constraint which is written as

$$\eta_{\Phi}(u,w) = \left\{ F_{\mathrm{in}}E_{ijkl}\varepsilon_{ij}(u_1)\varepsilon_{kl}(u_1) + F_{\mathrm{in}}\frac{2u_{1,\mathrm{out}}}{(1-k_s u_{2,\mathrm{out}})}E_{ijkl}\varepsilon_{ij}(u_1)\varepsilon_{kl}(u_2) + F_{\mathrm{in}}\frac{u_{1,\mathrm{out}}^2}{(1-k_s u_{2,\mathrm{out}})^2}E_{ijkl}\varepsilon_{ij}(u_2)\varepsilon_{kl}(u_2) \right\} \tag{55}$$

In numerical implementation, without remeshing, the strain energy density related to the shape gradient density can be accurately calculated the ''ersatz material'' approach [2], the geometry projection method [39] or the extended FE method [72]. This work adopts the simple ''ersatz material'' scheme to fill the void areas with one weak material. The element stiffness and the strain are then calculated under the assumption that they are proportional to their area-fractions of the solid material.

5. Mathematical programming approach

It is well known that the large-scale optimization with multiple constraints is rarely easy to implement [10]. Therefore, to efficiently solve the optimization problem of compliant mechanisms is really an important consideration. The optimality criteria method [48,80] have been successfully applied to solve problems with a large amount of design variables while single constraint, which usually includes the establishment of a heuristic updating scheme to renew the design variables. However, it is not always available to find an updating scheme for a complicated optimization problem [40], especially, a problem with a non-convex objective function [31], because the updating scheme is, more or less, a matter of experience. In addition, it is difficult to deal with multiple constrained optimization problems. However, several of the mathematical programming methods have been applied as more powerful and more theoretically well-established schemes to solve advanced structural optimization problems [54,55].

Svanberg [62] generalized the CONLIN approach [20] as the method of moving asymptotes (MMA) by introducing two sets of lower and upper asymptotes to dynamically adjust the convexity of the approximations. MMA, belonging to the sequential convex programming, has been regarded as one of the well-established algorithms for topology optimization problems [56,32]. Specifically, using the concept of the conservative convex separable approximations, the original optimization problem can be transformed into a set of linearized, convex and separable sub-problems based on the first or second derivatives at the current

and previous points. In particular, any of the sub-problems has a strictly convex objective function, which indicates a globally convergent optimum. At each step, the dual or the interior-point method can be used to renew design variables [64]. A new version of the globally convergent MMA (GCMMA) [65] has been developed as a more robust scheme in contrast to the previous versions which may fail on certain problems [62,63]. However, even if the newest version of GCMMA, it does not necessarily guarantee a globally convergent solution because the original problem itself is not convex [30] in despite of all sub-problems are strictly convex.

When GCMMA is used to solve the parameterization problem, an algorithm for selecting the move limit size is incorporated to stabilize the optimization [65]. However, as aforementioned, the move limit size here is different from the time-step size imposed by the CFL condition [52]. Therefore, the present method is expected to have a faster convergence, especially, for a finer mesh. In the conventional level set method, the methods for finding the Lagrange multiplier of the volume constraint appeared to be less effective. A fixed Lagrange multiplier [67] was introduced to make the mass conservative despite the fact that the constraint cannot be well satisfied during the entire optimization process. An alternative way for deciding the Lagrange multiplier [2] is under the assumption that the volume can remain unchanged during the front propagation, but actually the level set methods cannot make the volume conservative due to the drift of the volume during the optimization process. Wang et al. [75] applied the bisection algorithm to push the volume back, but it is difficult to handle multiple constraints. However, the GCMMA is very effective to handle multiple constraints [65] no matter they are globally or locally imposed. Thus, the present method is mass conservative.

According to GCMMA method, the original optimization formulation given in Eq. (32) should be further adjusted by introducing some artificial parameters [65] for the sake of ensuring a feasible initial design point, which is re-expressed as

$$\begin{cases} \underset{\boldsymbol{\alpha}=(\alpha_1,\alpha_2,\ldots,\alpha_n)^{\mathrm{T}}}{\text{Minimize}}: & J(\boldsymbol{\alpha}) + \left\{ a_0 z + \sum_{i=1}^{m} \left(c_i y_i + \frac{1}{2} d_i y_i^2 \right) \right\} \\ \text{Subject to}: & \begin{cases} G_i(\boldsymbol{\alpha}) - a_i z - y_i \leqslant g_i^*, \\ z \geqslant 0, y_i \geqslant 0, \\ \alpha_{j,\min} \leqslant \alpha_j \leqslant \alpha_{j,\max}, \\ \text{Equilibrium equations.} \end{cases} \end{cases} \tag{56}$$

where J is the objective function given in Eq. (32), and G_i $(i = 1, 2)$ are constraints. $\alpha_{j,\min}$ and $\alpha_{j,\max}$ are the lateral constraints of the design variables α_j. Here, the design variables are in fact the expansion coefficients defined in Eq. (18). $a_0 > 0$, $a_i \geqslant 0$, $c_i \geqslant 0$, $d_i \geqslant 0$ and $c_i + d_i > 0$ are the artificial parameters. $y = (y_1, y_2, \ldots, y_m)^T \in \mathbb{R}^m$ and $z \in \mathbb{R}$. m is the number of constraints and n is the number of design variables.

The sub-optimization problems related to the pervious formulation is defined in terms of GCMMA approximating expansions [65].

$$\begin{cases} \underset{\boldsymbol{\alpha}=(\alpha_1,\alpha_2,\ldots,\alpha_n)^{\mathrm{T}}}{\text{Minimize}}: & J^{(k)}(\boldsymbol{\alpha}) + \left(a_0 z + \sum_{i=1}^{m} (c_i y_i + \frac{1}{2} d_i y_i^2) \right) \\ \text{Subject to}: & \begin{cases} G_i^{(k)}(\boldsymbol{\alpha}) - a_i z - y_i \leqslant 0, \\ z \geqslant 0, y_i \geqslant 0, \\ \alpha_{j,\min} \leqslant \alpha_j \leqslant \alpha_{j,\max}, \\ \text{Equilibrium equations.} \end{cases} \end{cases} \tag{57}$$

where $J^{(k)}$ is the kth objective of sub-problems and $G_i^{(k)}$ represents the kth approximation of the ith constraint. Here, we only show the relevant optimization formulations. The details of selecting parameters can refer to Svanberg [62,65].

6. Numerical application

The synthesis of compliant inverter is one of the widely studied examples in the literature. Here, it is still presented to demonstrate the availability of the present parameterization level set method. First, it should be

addressed that the knots have a strong influence on the accuracy of the interpolation. In fact, CSRBF knots and FE nodes are two different sets of grids, and we can configure a denser or sparser CSRBF knots separately via other alternative schemes while keeping FE nodes unchanged. In this paper, CSRBF knots are supposed to be consistent with FE nodes to merely simplify the numerical procedure.

The level set function is initially embedded into the design domain as a signed distance function, but no re-initialization is applied further during the subsequent iterations. The material properties are defined with Young's modulus is 110 GPa and Poisson's ratio is 0.3. The volume ratio is limited to 0.25, while a maximum displacement 12 μm is prescribed at the input position to limit the maximal stress in the resulting mechanism. The terminal criterion is that the relative difference of two successive objective values is less than 0.001.

The design domain is defined as Fig. 8 with a domain size 240 μm × 240 μm, in which the left-upper and left-lower parts are fixed as the Dirichlet boundary. An actuation force $f_{\text{in}} = 500$ μN is applied at the center point of the left side (input port) as a non-homogenous Neumann boundary, and an artificial spring with stiffness $k_{\text{s}} = 0.2$ N/mm is attached at the output position to simulate the resistance from a work-piece.

First, only the lower half of the design domain is discretized with 7200 elements by fully considering its symmetry. The initial designs and the related level set surfaces with C4 function are given in Fig. 9. To demonstrate the influence of support radii on final designs, we apply the radius values as 1, 2.5, 5 and 20 times as the element length, respectively. The final topologies and the relevant level set surface for four different cases are shown in Fig. 10. The case (a) in Fig. 10 shows that the boundary of the final topology is not smooth enough, as a small support radius denotes a near singular matrix with a very small bandwidth, which leads to an unstable optimization process due to a larger numerical error, although the computational expense is cheaper (each CSRBF subroutine is 109.315s). As indicated by cases (b) and (c), increasing the support radius, we find that the smoothness of the design boundary is improved and the optimization process is becoming stable. The bandwidth of the matrix is slightly increased and computation effort is as a result slightly increased (each CSRBF subroutine is from 184.421 to 201.343s). The case (d) in Fig. 10 shows the design obtained with a

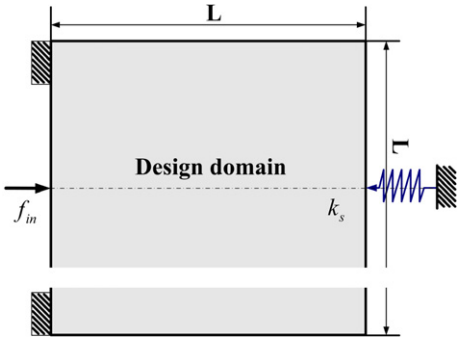


Fig. 8. Design domain of compliant micro-inverter.

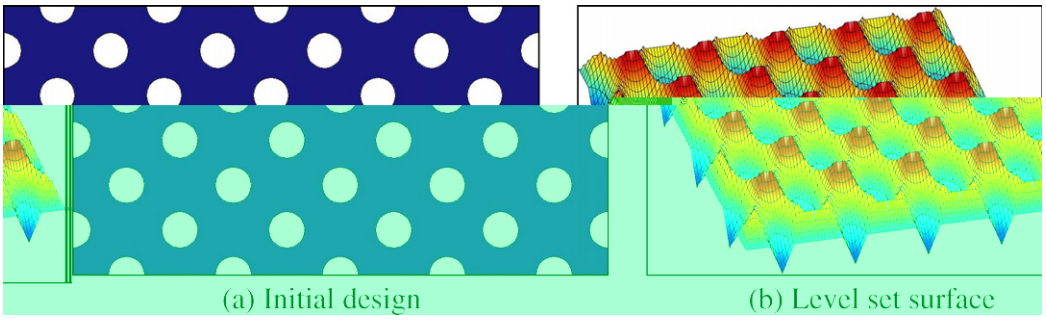


Fig. 9. Initial design and its level set surface.

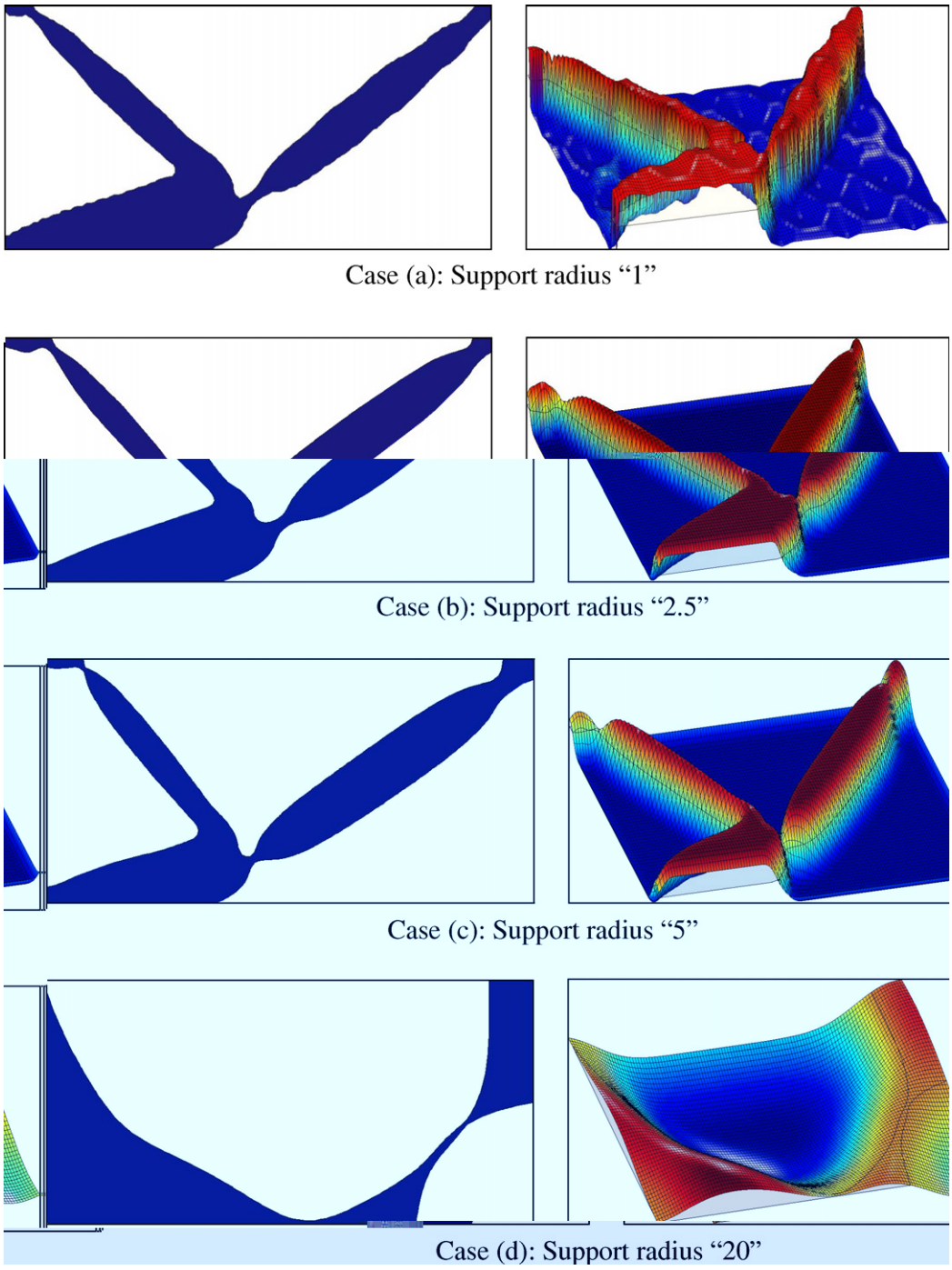


Fig. 10. Optimal topologies and level set surfaces with different support radii.

very large support radius (each CSRBF subroutine is 405.125s). As its numerical process shows, in addition to the computation effort being obviously increased, a too large support radius shows a strong diffusion term which makes the final topology is far beyond a widely recognized one. The experience criterion of choosing the support radius is to take into account the trade-off between the matrix errors and computation effects. Thus, the smoothness of the trial function and a moderate numerical efficiency can both be ensured [78]. In

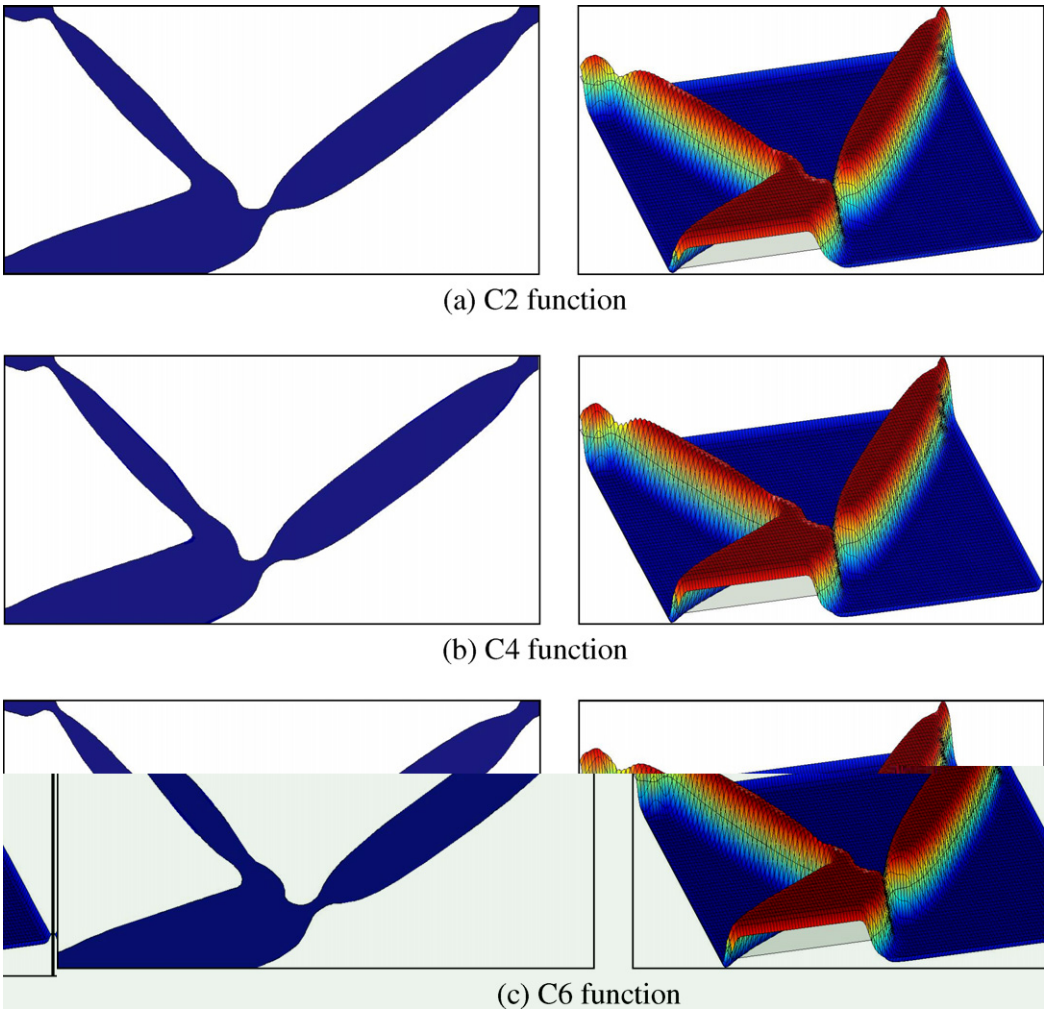


Fig. 11. Optimal topologies and the related level set surfaces.

terms of our experiences, the value of support radius is in general recommended as 2–4 times as the mesh size [19]. Furthermore, we implemented three cases using C2, C4 and C6 function with the same support radius 3, respectively. As displayed in Fig. 11, the final results are similar and it is hard to figure out the obvious difference. In fact, C2, C4, C6 or a variety of alternative CSRBFs can be available in approximating the level set surface with favorable smoothness. Here, the C4 function is only utilized as a demonstration in the rest of numerical cases.

Second, the entire domain is regarded as the design space discretized by 14,400 elements for displaying the whole mechanism. The iterative process of zero level set-contours is given in Fig. 12, and the related level set surfaces are shown in Fig. 13. It can be seen that all designs in different stages are characterized with a smooth design boundary, which means the present method can addresses the shape fidelity and topology changes simultaneously with a smooth boundary advection. It is well known that a smooth design boundary is especially meaningful for the shape optimization of boundary motion. The distributions of the expansion coefficients at different stages are shown in Fig. 14. We find that the design variables gradually gather towards two extreme-ends along with the optimization process. When the design approaches to the optimal point, almost all the design variables related to the weak material are swarmed to the same minimal value while those corresponding to the solid material reach to a maximal value. Hence, the changing process of the design variables here bears a similarity to that of the SIMP [55]. From the observation of the numerical process, it can be

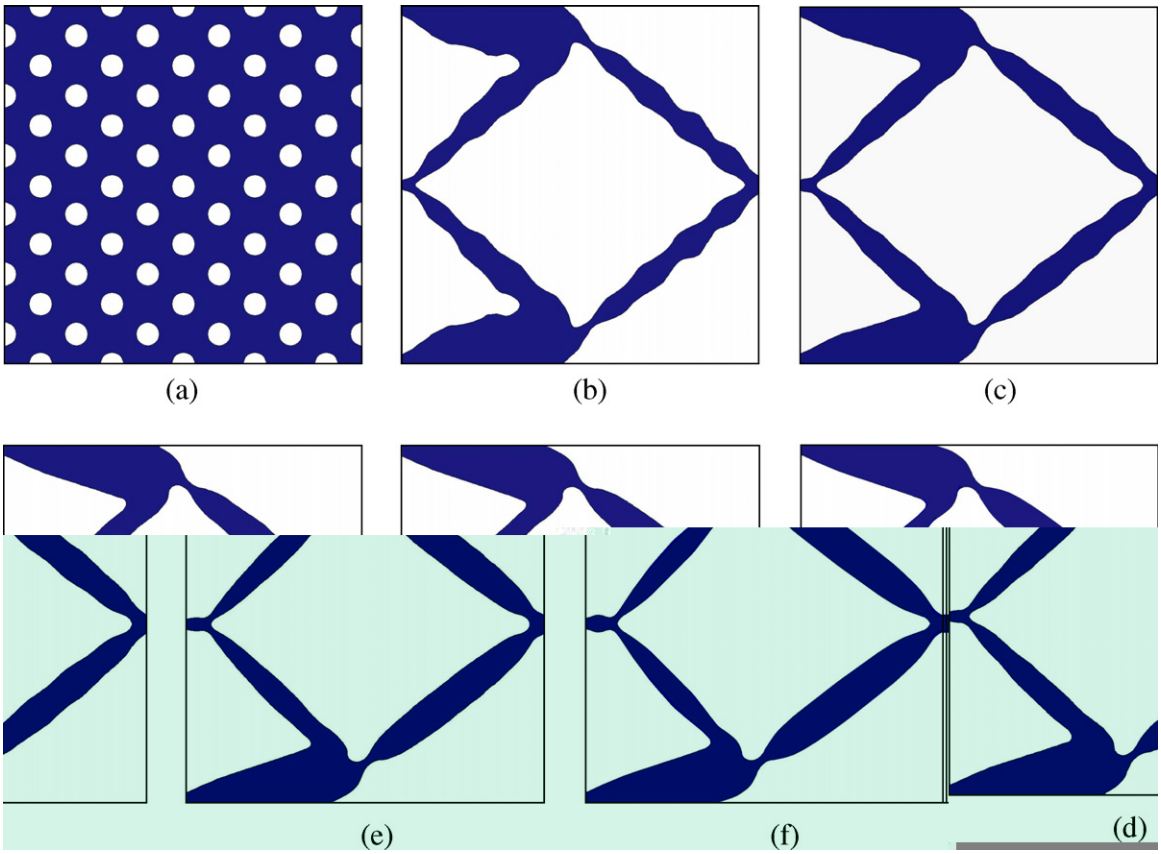


Fig. 12. Evolution of the topology related to zero-level set: (a) initial design; (b) step 65; (c) step 100; (d) step 150; (e) step 250 and (f) final design.

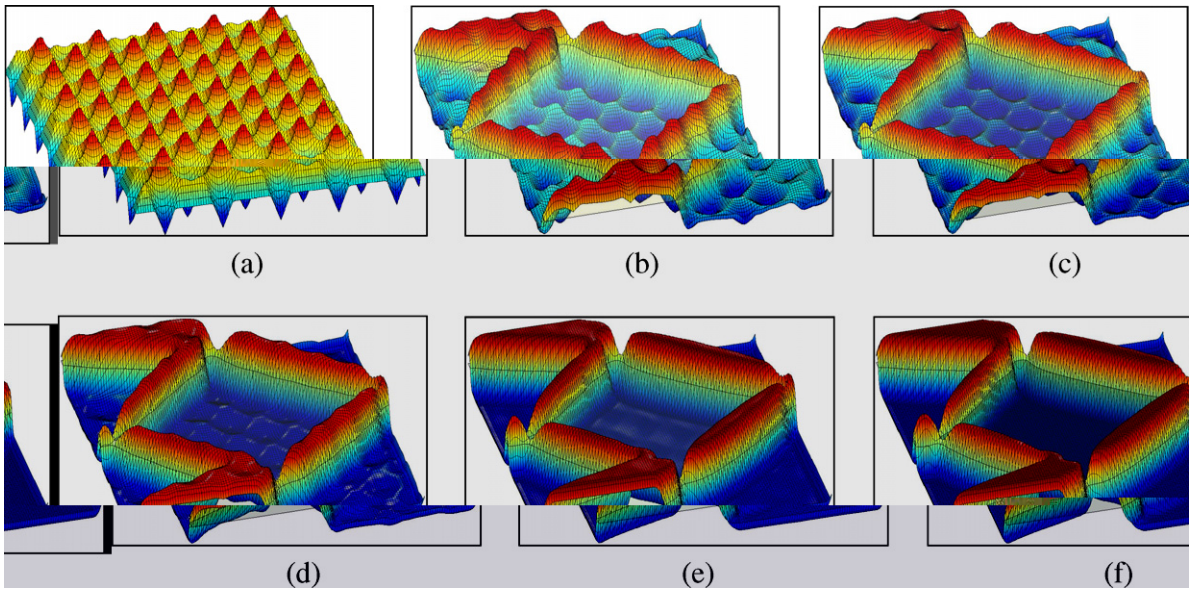


Fig. 13. Evolution of the level set surface.(a) initial surface; (b) step 65; (c) step 100; (d) step 150; (e) step 250 and (d) final surface.

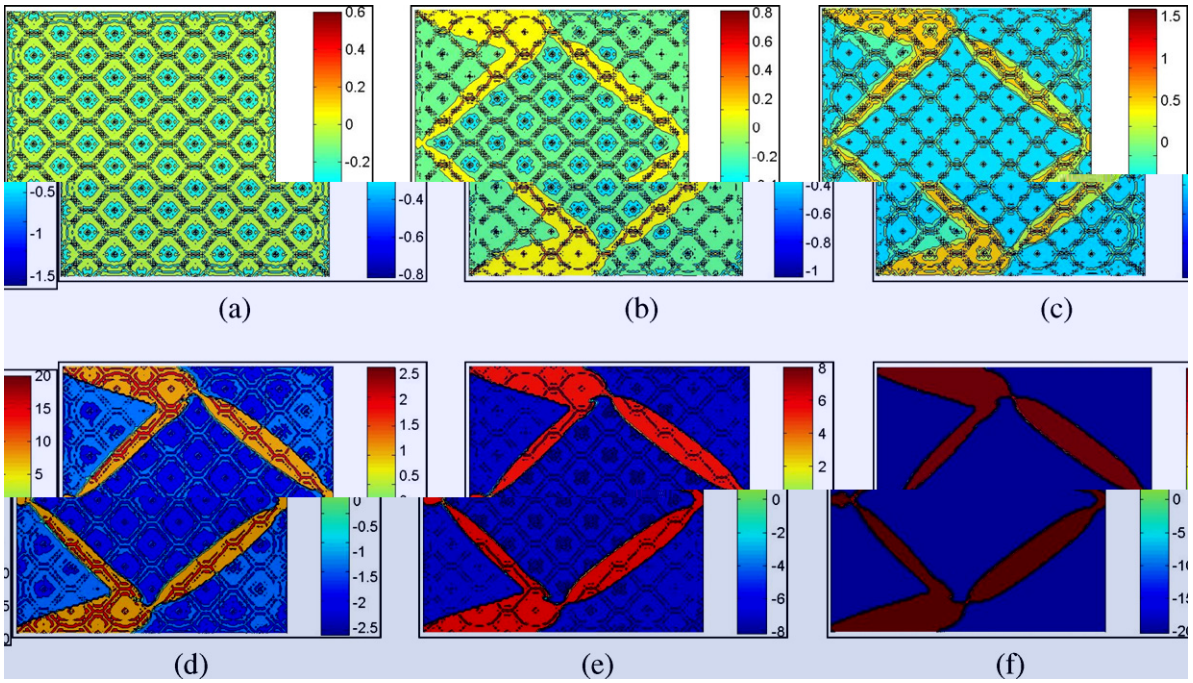


Fig. 14. Evolution of the expansion coefficients. (a) initial distribution; (b) step 65; (c) step 100; (d) step 150; (e) step 250 and (f) final distribution.

found that the shape gradient density of the optimal configuration distributes along the design boundary uniformly, which is theoretically consistent with the Kuhn–Tucker optimality condition [67]. Fig. 15 displays the curves of ME and the volume constraint over the iterations, where ME increases from 0.0317 to 0.6308 at start 70 iterations in which the topology optimization is almost completed. After that, it takes 322 iterations to continuously increase ME from 0.6308 to 0.7616. But these iterations are necessary to complete the shape optimization so as to ensure a uniform distribution of the strain energy density. Thus, in applying the present parameterization method to shape and topology optimization, the topology optimization is mainly used to

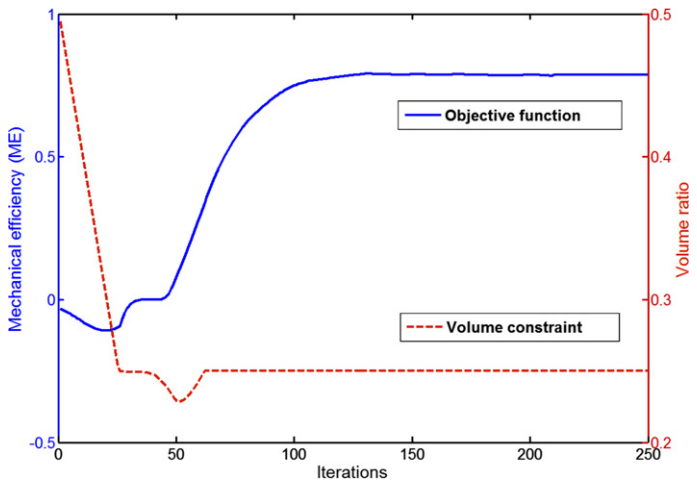


Fig. 15. Convergence histories for case with 7200 elements.

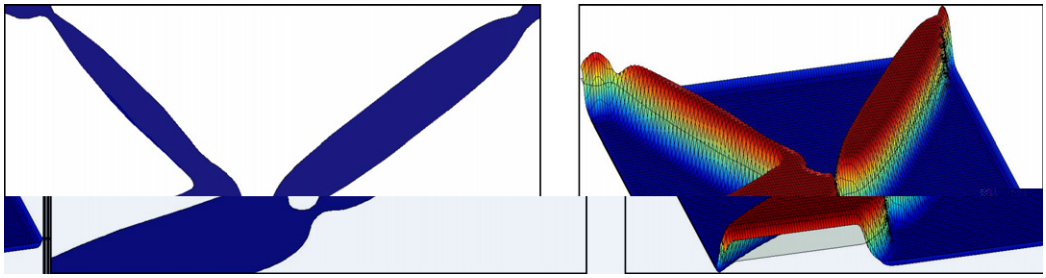


Fig. 16. Final topology and its level set surface with 9800 elements.

get an overall optimal layout while the shape optimization is applied to improve the specified local performances.

Third, to substantiate the present parameterization method free from the CFL condition, the lower half of the design domain is discretized with 9800 and 12,800 elements, respectively. In these cases, their initial designs are positioned with the same numbers of holes as configured in Fig. 9. For the first case with 9800 elements, the final design and the corresponding level set surface are given in Fig. 16, respectively. ME is maximized to 0.7681 after 391 iterations. Similarly, the final results of the second case with 12,800 elements are expressed in Fig. 17, and ME is obtained as 0.7704 after 384 iterations. In these two cases, the slight difference is a finer mesh more capable of describing the numerical FE model and its boundary condition. In terms of the final designs displayed in Figs. 12, 16 and 17, we can easily find that the similar designs can be obtained with the same move limits ($m = 0.05$), if the initial designs are discretized with different numbers of FE elements. However, in the conventional level set method [67], it is well known that the time-step size for an explicit time-marching scheme must be kept small enough to satisfy the CFL condition. Hence, for a finer FE mesh, the computation effort of the conventional level set method will be evidently increased because the time-step size is determined by the minimum grid size. If the same conditions are used to test the inverter design by using the conventional level set method [70], we find that the 9800 and 12,800 elements require more than 2100 and 2600 iterations for outcomes of the final designs, respectively. To address our major concern, the numerical details are omitted here. In the present method, the move limit has nothing to do with the meshing size and it can be flexibility adjusted by using MMA algorithm [65] to stabilize the optimization process. Hence, the present method is thoroughly free from the CFL condition and also it is time-stable for all CSRBF knots, which also explains the fact that the present algorithm has a rapid convergence to the final design. Further, if we increase the number of the CSRBF knots, theoretically, the boundary of the final design is prone to becoming smoother. However, it is unnecessary to overly increase the number of CSRBF knots, and it should be decided by the practical design because too many CSRBF knots can obviously influence the computational efficiency. Also, the extension velocity algorithm and the re-initialization procedure in the conventional level set method can also be reasonably avoided, as it is required to maintain the numerical accuracy of the Hamilton–Jacobi PDE. Hence, the present method can eliminate the numerical limits of the conventional level set method.

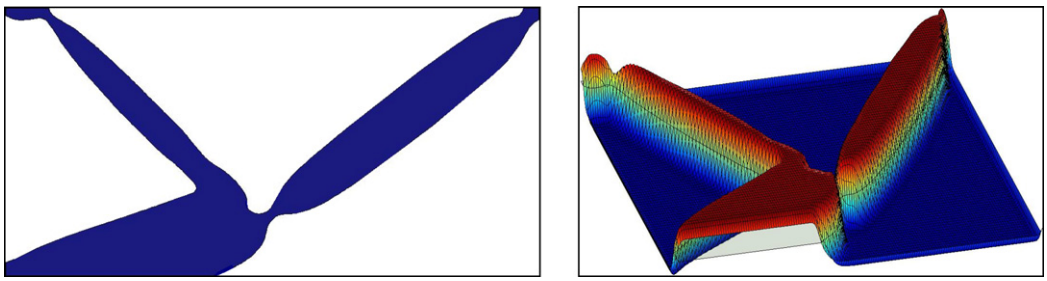
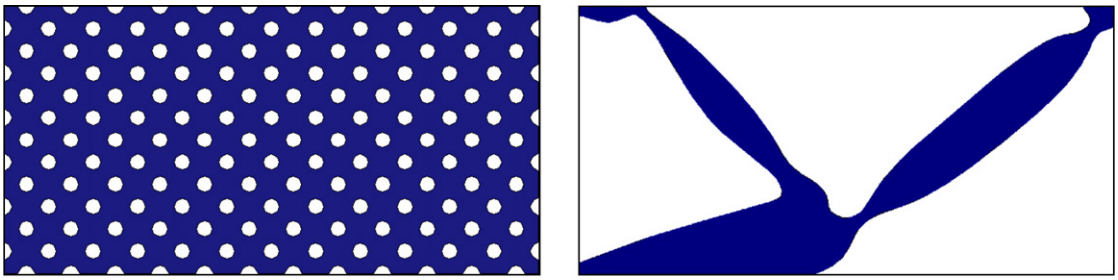
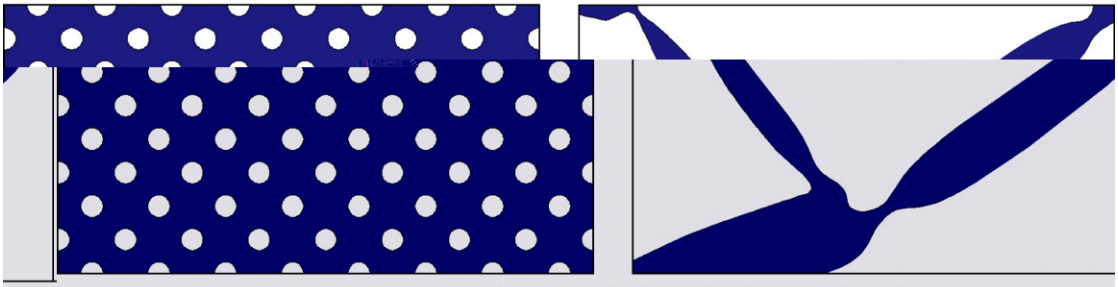


Fig. 17. Final topology and its level set surface with 12,800 elements.

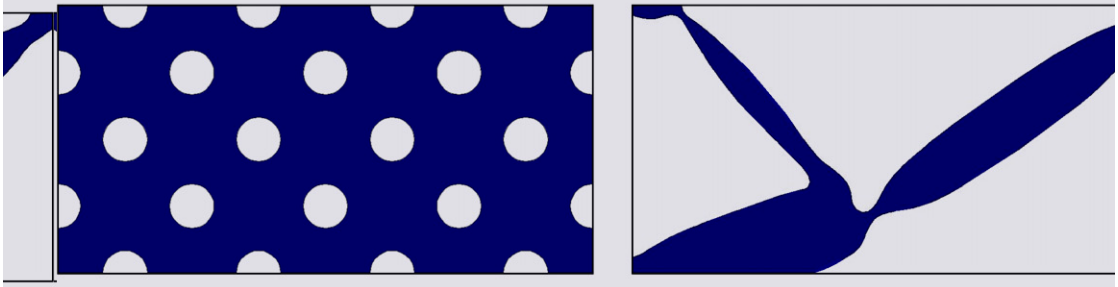
Finally, the influence of the different initializations upon the final results is also explored with the lower half of the structure discretized with 7200 elements. The cases in Fig. 18 display the optimal designs derived from the same domain initialized with different holes. The convergent iterations are 358, 341, 336 and 311, respectively, and the optimal solutions are all around 0.7000. It can be seen that the very similar topologies can be obtained even if the design domain is initialized with different holes. Hence, we conclude that the similar minima can be achieved as long as the holes positioned in the initial design are enough to completely describe all



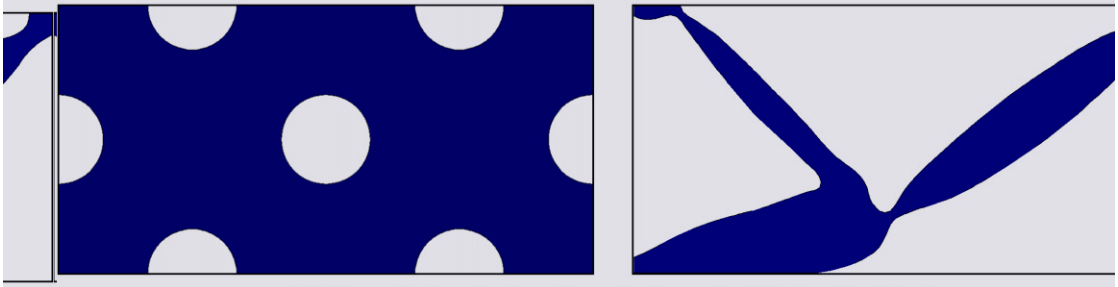
(a) Case 1: the design domain is embedded with 126 holes



(b) Case 2: the design domain is embedded with 52 holes



(c) Case 4: the design domain is embedded with 10 holes



(d) Case 5: the design domain is embedded with 1 hole

Fig. 18. Topologies with different initial holes.

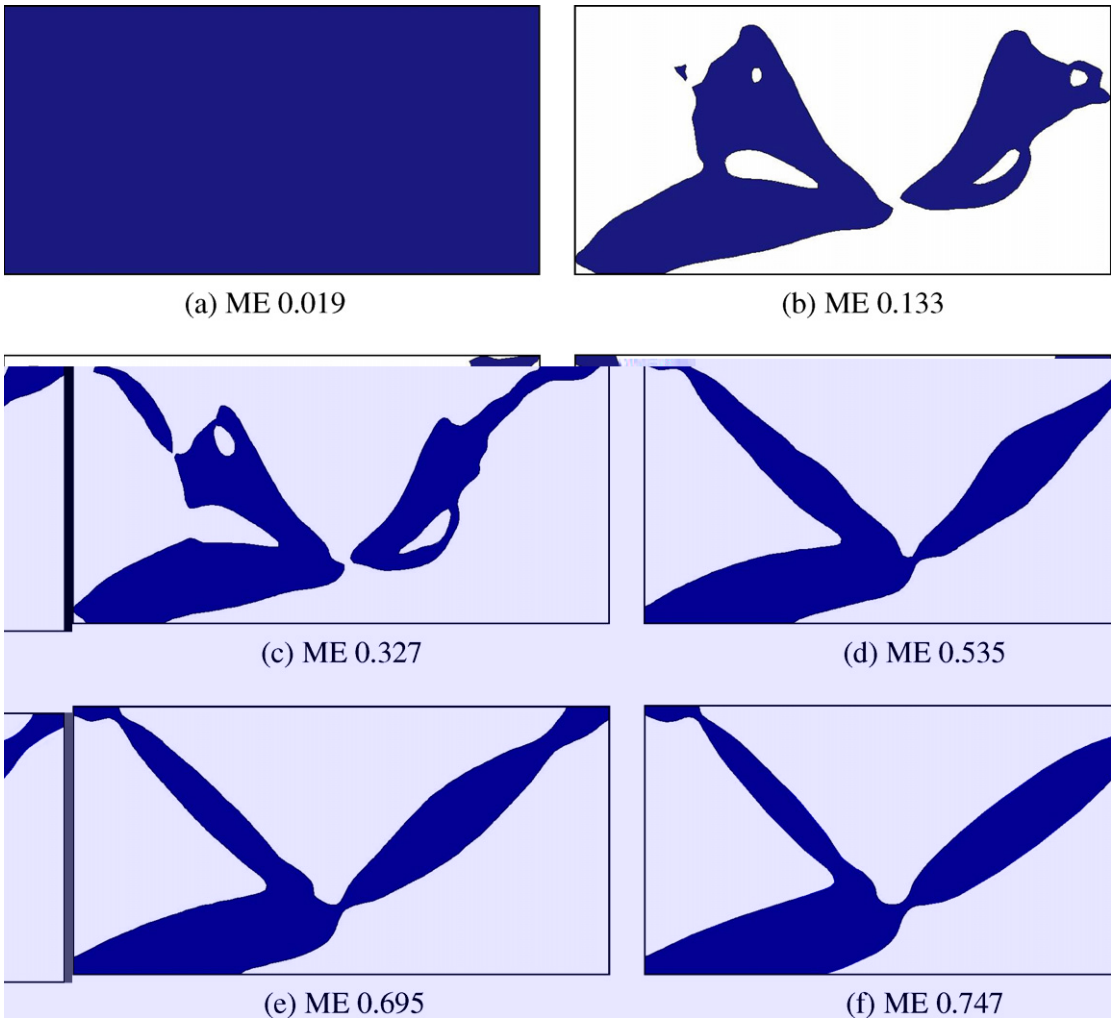


Fig. 19. Evolution of the topology without any hole: (a) initial design; (b) step 25; (c) step 50; (d) step 100; (e) step 200 and (f) final design.

admissible shape configurations. The simple way to foretell an initial topology can be in conformity to Allaire et al. [2]. To further show the capability of the present method, the design domain is initialized without any hole, which is obviously incapable of describing a recognized topology. But Fig. 19 clearly displays that the present method can generate new holes inside the design domain automatically, in which Fig. 19a is the initial design and Fig. 19f is the final design obtained after 327 iterations. The corresponding mechanical efficiencies (MEs) related to the objective function are also given in Fig. 19. In the conventional level set method, the creation of new holes is not allowed inside the design domain. Therefore, the topological derivative method [4,13] is usually included to make the following design less dependent on the initial guess or the topological derivative [13] method is usually included to make the final design less dependent on the initial guess. By eliminating the global re-initialization procedure in the conventional level set method [67], the present parameterization method shows its capacity of ''nucleating'' new holes inside the material domain with the aid of the nature velocity extension.

7. Conclusions

This paper has presented a parameterization level set method for shape and topology optimization of compliant mechanisms by using compactly supported radial basis functions. The structural boundary is implicitly

represented as a level set model by implicitly embedding into a higher-dimensional level set function which is further interpolated using the CSRBF. By discretizing the Hamilton–Jacobi PDE into a set of algebraic equations, the original shape and topology optimization is transformed into a size optimization of the expansion coefficients to which the MMA method is applied. Those unfavorable features on the temporal and spatial discretizations of the Hamilton–Jacobi PDE are now totally eliminated. The optimization is then changed to a process of moving structural boundary by finding the expansion coefficients to approximate the level set function in conjunction with CSRBFs located on the knots. The widely studied inverter mechanism is used to demonstrate the present method.

The present method has provided a new possibility for meaningfully combining the level set method with the well-established optimization methods. With the parameterization scheme devised in this work, we can fully take advantages of both the implicit level set representation and the well-founded optimization algorithm. In particular, the present method is capable of dealing with advanced shape and topology optimization associated with a complicated objective function and multiple constraints, globally or locally imposed. It should be noted that the present method cannot physically ensure the elimination of de-facto hinges which is now still an open topic in this area [10]. A complete investigation is therefore outside of the scope of this paper. The method recently presented by Chen and Wang [17] might be an alternative scheme to handle hinges in design of compliant mechanisms.

Acknowledgments

The research work sponsored in part by the Australian Research Council under Grant No. ARC-DP0666683, also by the Hong Kong Research Grants Council (Grants: CUHK4164/03E and CUHK416205) and the Natural Science Foundation of China (Grants: 50128503 and 50390063). The authors show their acknowledgements to Prof. Krister Svanberg for providing his MMA codes, and to the anonymous reviewers for their valuable comments.

Appendix A

The aim of this section is to develop the shape derivative of $J_\Phi(u)$ in Eq. (37). First, taking the Fréchet derivative of $a_\Phi(u,\upsilon)$ with respect to Φ in the direction of γ, we have

$$\left\langle \frac{\mathrm{d}a_\Phi(u,\upsilon)}{\mathrm{d}\Phi}, \gamma \right\rangle = \left\langle \frac{\partial a_\Phi(u,\upsilon)}{\partial u}, \delta u \right\rangle + \left\langle \frac{\partial a_\Phi(u,\upsilon)}{\partial \Phi}, \gamma \right\rangle \tag{A.1}$$

where the two terms on the right-hand side can be separately expressed as

$$\left\langle \frac{\partial a_\Phi(u,\upsilon)}{\partial u}, \delta u \right\rangle = a_\Phi(\delta u, \upsilon) \tag{A.2}$$

$$\left\langle \frac{\partial a_\Phi(u,\upsilon)}{\partial \Phi}, \gamma \right\rangle = \int_D E_{ijkl}\varepsilon_{ij}(u)\varepsilon_{kl}(\upsilon)\delta(\Phi)\gamma\,\mathrm{d}\Omega \tag{A.3}$$

where δu denotes the variation of u with respect to Φ.

The Fréchet derivative of $l_\Phi(\upsilon)$ with respect to Φ in the direction of γ is given as

$$\left\langle \frac{\partial l_\Phi(\upsilon)}{\partial \Phi}, \gamma \right\rangle = \int_D p\upsilon\delta(\Phi)\gamma\,\mathrm{d}\Omega + \int_D \mathrm{div}\left(\tau\upsilon\frac{\nabla\Phi}{|\nabla\Phi|}\right)\delta(\Phi)\gamma\,\mathrm{d}\Omega + \int_{\partial D}\tau\upsilon\frac{\delta(\Phi)}{|\nabla\Phi|}\frac{\partial\Phi}{\partial n}\gamma\,\mathrm{d}\Gamma \tag{A.4}$$

With the Fréchet derivatives in (A.1) and (A.2), the state equation in (38) can be specified as

$$a_\Phi(\delta u, \upsilon) = \left\langle \frac{\partial l_\Phi(\upsilon)}{\partial \Phi}, \gamma \right\rangle - \left\langle \frac{\partial a_\Phi(u,\upsilon)}{\partial \Phi}, \gamma \right\rangle \tag{A.5}$$

The Frechet derivative of $J_\Phi(u)$ with respect to Φ in the direction of γ is written as

$$\left\langle \frac{\mathrm{d}J_\Phi(u)}{\mathrm{d}\Phi}, \gamma \right\rangle = \left\langle \frac{\partial J_\Phi(u)}{\partial u}, \delta u \right\rangle + \left\langle \frac{\partial J_\Phi(u)}{\partial \Phi}, \gamma \right\rangle \tag{A.6}$$

where the right-hand side terms are given, respectively, as

$$\left\langle \frac{\partial J_{\Phi}(u)}{\partial u}, \delta u \right\rangle = \int_D \frac{\partial \vartheta(u)}{\partial u} \delta u H(\Phi) \mathrm{d}\Omega \tag{A.7}$$

$$\left\langle \frac{\partial J_{\Phi}(u)}{\partial \Phi}, \gamma \right\rangle = \int_D \vartheta(u) \delta(\Phi) \gamma \, \mathrm{d}\Omega \tag{A.8}$$

To simplify the Fréchet derivative of $J_{\Phi}(u)$, an adjoint displacement field w is introduced to eliminate the intermediate variable δu, leading to the following adjoint equation

$$a_{\Phi}(\upsilon, w) = \int_D \frac{\partial \vartheta(u)}{\partial u} H(\Phi) \upsilon \, \mathrm{d}\Omega \quad \forall \upsilon \in \mathbf{V}, \quad w|_{\Gamma_D} = 0 \tag{A.9}$$

Let υ in the state equilibrium equation (A.5) be w, we have

$$a_{\Phi}(\delta u, w) = \left\langle \frac{\partial l_{\Phi}(w)}{\partial \Phi}, \gamma \right\rangle - \left\langle \frac{\partial a_{\Phi}(u, w)}{\partial \Phi}, \gamma \right\rangle \tag{A.10}$$

At the same time, substitution of $\upsilon = \delta u$ into the adjoint equation (A.9) can yield

$$a_{\Phi}(\delta u, w) = \int_D \frac{\partial \vartheta(u)}{\partial u} H(\Phi) \delta u \, \mathrm{d}\Omega \tag{A.11}$$

Then, by using Eqs. (A.7), (A.10) and (A.11), we have the following Eq. (A.12)

$$\int_D \frac{\partial \vartheta(u)}{\partial u} H(\Phi) \delta u \, \mathrm{d}\Omega = \left\langle \frac{\partial J_{\Phi}(u)}{\partial u}, \delta u \right\rangle = \left\langle \frac{\partial l_{\Phi}(w)}{\partial \Phi}, \gamma \right\rangle - \left\langle \frac{\partial a_{\Phi}(u, w)}{\partial \Phi}, \gamma \right\rangle \tag{A.12}$$

Letting υ ($\upsilon \in \mathbf{V}$) in Eqs. (A.3) and (A.4) be w ($w \in \mathbf{V}$), and then substituting them into Eq. (A.12), Eq. (A.7) can then be re-expressed as follows:

$$\left\langle \frac{\partial J_{\Phi}(u)}{\partial u}, \delta u \right\rangle = \int_D \left(pw + \mathrm{div}\left(\tau w \frac{\nabla \Phi}{|\nabla \Phi|} \right) - E_{ijkl} \varepsilon_{ij}(u) \varepsilon_{kl}(w) \right) \delta(\Phi) \gamma \, \mathrm{d}\Omega + \int_{\partial D} \tau w \frac{\delta(\Phi)}{|\nabla \Phi|} \frac{\partial \Phi}{\partial n} \gamma \, \mathrm{d}\Gamma \tag{A.13}$$

According to Eqs. (A.13), (A.8) and (A.6), we can get the Fréchet derivative of $J_{\Phi}(u)$

$$\left\langle \frac{\mathrm{d}J_{\Phi}(u)}{\mathrm{d}\Phi}, \gamma \right\rangle = \int_D \left(\vartheta(u) + pw + \mathrm{div}\left(\tau w \frac{\nabla \Phi}{|\nabla \Phi|} \right) - E_{ijkl} \varepsilon_{ij}(u) \varepsilon_{kl}(w) \right) \delta(\Phi) \gamma \, \mathrm{d}\Omega + \int_{\partial D} \tau w \frac{\delta(\Phi)}{|\nabla \Phi|} \times \frac{\partial \Phi}{\partial n} \gamma \, \mathrm{d}\Gamma \tag{A.14}$$

The normal velocity field $v_n = \mathrm{v}(x) \cdot n(\mathrm{x})$ and the normal velocity operator, respectively, satisfy the following conditions

$$\frac{\partial \Phi}{\partial t} = \mathrm{v}_n |\nabla \Phi|, \quad \left. \frac{\partial \Phi}{\partial n} \right|_{\partial D} = 0 \tag{A.15}$$

Now, in terms of Eqs. (A.13), (A.14) and (A.15), the derivative of $J_{\Phi}(u)$ with respect to pseudo-time t can be constructed as follows:

$$\begin{aligned} \frac{\mathrm{d}J_{\Phi}(u)}{\mathrm{d}t} &= \left\langle \frac{\partial J_{\Phi}(u)}{\partial u}, \frac{\partial \Phi}{\partial t} \right\rangle + \left\langle \frac{\partial J_{\Phi}(u)}{\partial \Phi}, \frac{\partial \Phi}{\partial t} \right\rangle \\ &= \int_D \mathrm{v}_n \left(\vartheta(u) + pw + \mathrm{div}\left(\tau w \frac{\nabla \Phi}{|\nabla \Phi|} \right) - E_{ijkl} \varepsilon_{ij}(u) \varepsilon_{kl}(w) \right) \delta(\Phi) |\nabla \Phi| \mathrm{d}\Omega \end{aligned} \tag{A.16}$$

The shape derivative $J'(u)$ in Eq. (A.16) indicates that a variation in shape Φ can have an influence on both the function $J(u)$ and the state variable u.

References

[1] G. Allaire, F. Jouve, optimal design of micro-mechanisms by the homogenization method, European Journal of Finite Element 11 (2002) 405–416.
[2] G. Allaire, F. Jouve, A.M. Toader, Structural optimization using sensitivity analysis and a level-set method, Journal of Computational Physics 194 (1) (2004) 363–393.
[3] G. Allaire, F. Jouve, A level-set method for vibration and multiple loads structural optimization, Computer Methods in Applied Mechanics and Engineering 194 (30–33) (2005) 3269–3290.
[4] G. Allaire, F. Jouve, Coupling the level set method and the topological gradient in structural optimization, in: M.P. Bendsoe, N. Olhoff, O. Sigmund (Eds.), IUTAM Symposium on Topological Design Optimization of Structures, Machines and Materials, Springer, Netherlands, 2006, pp. 3–12.
[5] G.K. Ananthasuresh, S. Kota, Y. Gianchandani, A methodical approach to the design of compliant micro-mechanisms, in: Solid-State Sensor and Actuator Workshop, 1994, pp. 89–192.
[6] G.K. Ananthasuresh, Optimal Synthesis Methods for MEMS, Kluwer, Boston, 2003.
[7] T. Belytschko, S.P. Xiao, C. Parimi, Topology optimization with implicitly function and regularization, International Journal for Numerical Method in Engineering 57 (2003) 1177–1196.
[8] M.P. Bendsøe, N. Kikuchi, Generating optimal topology in structural design using a homogenization method, Computer Methods in Applied Mechanics and Engineering 71 (2) (1988) 197–224.
[9] M.P. Bendsøe, O. Sigmund, Material interpolation schemes in topology optimization, Archive of Applied Mechanics 69 (1999) 635–654.
[10] M.P. Bendsøe, O. Sigmund, Topology Optimization: Theory, Methods, and Applications, Springer, Berlin, Heidelberg, 2003.
[11] T.E. Bruns, D.A. Tortorelli, Topology optimization of nonlinear elastic structures and compliant mechanisms, Computer Methods in Applied Mechanics and Engineering 190 (26–27) (2001) 3443–3459.
[12] M.D. Buhmann, Radial Basis Functions: Theory and Implementations, Cambridge Monographs on Applied and Computational Mathematics, vol. 12, Cambridge University Press, NY, 2004.
[13] M. Burger, B. Hacker, W. Ring, Incorporating topological derivatives into level set methods, Journal of Computational Physics 194 (2004) 344–362.
[14] S. Canfield, M. Frecker, Topology optimization of compliant mechanical amplifiers for piezoelectric actuators, Structural and Multidisciplinary Optimization 20 (2000) 269–279.
[15] T. Cecil, J.L. Qian, S. Osher, Numerical methods for high dimensional Hamilton–Jacobi equations using radial basis functions, Journal of Computational Physics 196 (1) (2004) 327–347.
[16] K.K. Choi, N.H. Kim, Structural Sensitivity Analysis and Optimization I-Linear Systems, Springer, New York, 2005.
[17] S.K. Chen, M.Y. Wang, Geometric width control in topology optimization using level set method and a quadratic energy functional, in: ASME Proceedings of IDETC/CIE 2006, September 10–13, 2006, Philadelphia, Pennsylvania, USA.
[18] M.G. Crandall, P.L. Lions, Viscosity solutions of Hamilton–Jacobi equations, Transact American Mathematics Society 277 (1983) 1–43.
[19] J. Dolbow, T. Belytschko, An introduction to programming the meshless element free Galerkin method, Archives of Computational Methods in Engineering 5 (3) (1998) 207–241.
[20] C. Fleury, CONLIN: an efficient dual optimizer based on convex approximation concepts, Structural and Multidisciplinary Optimization 1 (1989) 81–89.
[21] M.I. Frecker, G.K. Ananthasuresh, S. Nishiwaka, N. Kikuchi, S. Kota, Topological synthesis of compliant mechanisms using multi-criteria optimization, Journal of Mechanical Design 119 (2) (1997) 238–245.
[22] M.I. Frecker, Recent advances in optimization of smart structures and actuators, Journal of Intelligent Material Systems and Structures 14 (2003) 207–216.
[23] E. Haber, A multilevel, level-set method for optimizing eigenvalues in shape design problems, Journal of Computational Physics 198 (2) (2004) 518–534.
[24] S. Kota, J. Joo, Z. Li, Design of compliant mechanisms: applications to MEMS, Analog Integrated Circuits and Signal Processing – An International Journal 29 (2001) 7–15.
[25] S. Kota, K.J. Lu, Z. Kreiner, B. Trease, J. Arenas, J. Geiger, Design and application of compliant mechanisms for surgical tools, Journal of Biomechanical Engineering 127 (6) (2005) 981–989.
[26] L.L. Howell, A. Midha, T.W. Norton, Evaluation of equivalent spring stiffness for use in a pseudo-rigid-body model of large-deflection compliant mechanisms, Journal of Mechanical Design 118 (1) (1996) 126–131.
[27] L.L. Howell, Compliant Mechanisms, John Wiley & Sons, Inc., New York, 2001.
[28] L.D.D. Lageneste, H. Pitsch, A Numerical Scheme for the Large-eddy Simulation of Turbulent Combustion using a Level-set Method, Annual Research Briefs, Center for Turbulence Research, Stanford University, Stanford, CA, USA, 2002.
[29] U.D. Larsen, O. Sigmund, S. Bouwstra, Design and fabrication of compliant micromechanisms and structures with negative Poisson's ratio, Journal of Microelectromechanical System 6 (2) (1997) 99–106.
[30] G.K. Lau, H. Du, M.K. Lim, Use of functional specifications as objective functions in topological optimization of compliant mechanism, Computer Methods in Applied Mechanics and Engineering 190 (2001) 4421–4433.
[31] G.K. Lau, H. Du, M.K. Lim, Convex analysis for topology optimization of compliant mechanisms, Structural and Multidisciplinary Optimization 22 (4) (2001) 284–294.

[32] Z. Luo, L.P. Chen, J.Z. Yang, Y.Q. Zhang, K. Abdel-Malek, Compliant mechanism design using multi-objective topology optimization scheme of continuum structures, Structural and Multidisciplinary Optimization 30 (2) (2005) 142–154.
[33] G.S. Jiang, D. Peng, Weighted ENO schemes for Hamilton–Jacobi equations, SIAM Journal on Scientific Computing 21 (2000) 2126–2143.
[34] A. Midha, L.L. Howell, T.W. Norton, Limit positions of compliant mechanisms using the pseudo-rigid-body model concept, Mechanism and Machine Theory 35 (2000) 99–115.
[35] I.M. Mitchell, A toolbox of level set methods, Technical Report, TR-2004-09, Department of Computer Science, University of British Columbia, Canada, 2004.
[36] T. Moulton, G.K. Ananthasuresh, Design and manufacture of electro-thermal-compliant micro-devices, Sensors and Actors, Series A: Physical 90 (2001) 38–48.
[37] D. Nathan, L.L. Howell, A self-retracted fully compliant bistable micromechanism, Journal of Microelectromechanical Systems 12 (3) (2003) 273–280.
[38] J. Nocedal, S.J. Wright, Numerical Optimization, Springer, New York, 1999.
[39] J. Norato, J.R. Haber, D. Tortorelli, M.P. Bendsoe, A geometry projection method for shape optimization, International Journal for Numerical Methods in Engineering 60 (14) (2004) 2289–2312.
[40] S. Nishiwaki, M.I. Frecker, S. Min, N. Kikuchi, Topology optimization of compliant mechanisms using the homogenization method, International Journal for Numerical Methods in Engineering 42 (1998) 535–559.
[41] S. Osher, R.P. Fedkiw, Level set methods: an overview and some recent results, Journal of Computational Physics 169 (2) (2001) 463–502.
[42] S. Osher, R.P. Fedkiw, Level Set Methods and Dynamic Implicit Surface, Springer-Verlag, New York, 2002.
[43] S. Osher, F. Santosa, Level-set methods for optimization problem involving geometry and constraints: I. Frequencies of a two-density inhomogeneous drum, Journal of Computational Physics 171 (1) (2001) 272–288.
[44] S. Osher, J.A. Sethian, Front propagating with curvature dependent speed: algorithms based on Hamilton–Jacobi formulations, Journal of Computational Physics 78 (1988) 12–49.
[45] C.B.W. Pedersen, T. Buhl, O. Sigmund, Topology synthesis of large displacement compliant mechanisms, International Journal for Numerical Methods in Engineering 50 (2001) 2683–2705.
[46] D. Peng, B. Merriman, S. Osher, H. Zhao, M. Kang, A PDE-based fast local level set method, Journal of Computational Physics 155 (1999) 410–438.
[47] S. Rippa, An algorithm for selecting a good value for the parameter c in radial basis functions interpolation, Advances in Computational Mathematics 11 (2–3) (1999) 193–210.
[48] G.I.N. Rozvany, U. Kirsch, M.P. Bendsøe, O. Sigmund, Layout optimization of structures, Applied Mechanics Reviews 48 (2) (1995) 41–119.
[49] R. Schaback, H. Wendland, Using compactly supported radial basis functions to solve partial differential equations, in: C.S. Chen, C.A. Brebbia, D.W. Pepper (Eds.), Boundary Element Technology XIII, WitPress, Southampton, Boston, 1999, pp. 311–324.
[50] R. Schaback, H. Wendland, Characterization and construction of radial basis functions, in: N. Dyn, D. Leviatan, A. Pinkus (Eds.), Multivariate Approximation and Applications, Cambridge University Press, Cambridge, UK, 2001, pp. 1–24.
[51] R. Schaback, H. Wendland, Inverse and saturation theorems for radial basis function interpolation, Mathematics of Computation 71 (2002) 669–681.
[52] J.A. Sethian, Level Set Methods and Fast Marching Methods: Evolving Interfaces in Computational Geometry, Fluid Mechanics, Computer Version and Material Science, Cambridge Monograph on Applied and Computational Mathematics, Cambridge University Press, Cambridge, UK, 1999.
[53] J.A. Sethian, A. Wiegmann, Structural boundary design via level set and immersed interface methods, Journal of Computational Physics 163 (2000) 489–528.
[54] O. Sigmund, On the design of compliant mechanisms using topology optimization, Mechanics of Structures and Machines 25 (4) (1997) 493–524.
[55] O. Sigmund, Design of multiphysics actuator using topology optimization – part I: one material structure, Computer Methods in Applied Mechanics and Engineering 190 (49–50) (2001) 6577–6604.
[56] O. Sigmund, Design of multiphysics actuator using topology optimization – part II: two material structure, Computer Methods in Applied Mechanics and Engineering 190 (49–50) (2001) 6605–6627.
[57] O. Sigmund, J. Petersson, Numerical instabilities in topology optimization: a survey on procedures dealing with checkerboards, mesh-dependencies and local minima,, Structural and Multidisciplinary Optimization 16 (1998) 68–75.
[58] E.C.N. Silva, N. Kikuchi, Design of piezoelectric transducers using topology optimization, Smart Materials and Structures 8 (3) (1999) 350–364.
[59] J. Sokolowski, J.P. Zolesio, Introduction to shape optimization: shape sensitivity analysis, Springer, Berlin, 1992.
[60] J. Sokolowski, A. Zochowski, On the topological derivative in shape optimization, SIAM Journal of Control Optimization 37 (1999) 1251–1272.
[61] M. Sussman, E. Fatemi, An efficient interface-preserving level set redistancing algorithm and its application to interfacial incompressible fluid flow, SIAM Journal of Science Computing 20 (1999) 1165–1191.
[62] K. Svanberg, The method of moving asymptotes: a new method for structural optimization, International Journal for Numerical Method in Engineering 24 (1987) 359–373.
[63] K. Svanberg, A globally convergent version of MMA without linesearch, in: N. Olhoff, G. Rozvany (Eds.), Proceeding of the First World Congress of Structural and Multidisciplinary Optimization, Pergamon Press, Elmsford, NY, 1995, pp. 9–16.

[64] K. Svanberg, Two-primal-dual interior-point methods for the MMA subproblems, Technical Report TRITA-MAT-1998-OS12, Department of Mathematics, KTH, Stockholm, 1998.
[65] K. Svanberg, A class of globally convergent optimization methods based on conservative convex separable approximations, SIAM Journal of Optimization 12 (2) (2002) 555–573.
[66] R. Tsai, S. Osher, Level set methods and their applications in image science, Communications in Mathematical Sciences 1 (4) (2003) 623–656.
[67] M.Y. Wang, X.M. Wang, D.M. Guo, A level set method for structural topology optimization, Computer Methods in Applied Mechanics and Engineering 192 (2003) 224–246.
[68] M.Y. Wang, X.M. Wang, PDE-driven, level sets, shape sensitivity and curvature flow for structural topology optimization, Computer Modeling in Engineering and Science 6 (4) (2004) 373–396.
[69] M.Y. Wang, X.M. Wang, 'Color' level sets: a multiphase method for structural topology optimization with multiple materials, Computer Methods in Applied Mechanics and Engineering 193 (2004) 469–496.
[70] M.Y. Wang, S.K. Chen, X.M. Wang, Y.L. Mei, Design of multi-material compliant mechanisms using level set methods, Journal of Mechanical Design 127 (5) (2005) 941–956.
[71] P. Wei, M.Y. Wang, Parametric structural shape and topology optimization method with radial basis functions and level-set method, in: Proceedings of ASME, IDETC/CIE, 2006, Philadelphia, USA.
[72] S.Y. Wang, M.Y. Wang, A moving superimposed finite element method for structural topology optimization, International Journal for Numerical Method in Engineering 65 (11) (2006) 1892–1922.
[73] S.Y. Wang, M.Y. Wang, Radial basis functions and level set method for structural topology optimization, International Journal for Numerical Method in Engineering 65 (12) (2006) 2060–2090.
[74] S.Y. Wang, M.Y. Wang, Structural shape and topology optimization using an implicit free boundary parameterization method, Computer Modeling in Engineering and Science 13 (2) (2006) 119–147.
[75] S.Y. Wang, K.M. Lim, B.C. Khoo, M.Y. Wang, An extended level set method for shape and topology optimization, Journal of Computational Physics 221 (1) (2007) 395–421.
[76] H. Wendland, Piecewise polynomial, positive definite and compactly supported radial functions of minimal degree, Advances in Computational Mathematics 4 (1995) 389–396.
[77] H. Wendland, Error estimates for interpolation by compactly supported radial functions of minimal degree, Journal of Approximation Theory 93 (1998) 258–272.
[78] J.R. Xiao, Local heaviside weighted MLPG meshless method for two-dimensional solids using compactly supported radial basis functions, Computer Methods in Applied Mechanics and Engineering 194 (2004) 117–138.
[79] L. Yin, G.K. Ananthasuresh, Design of distributed compliant mechanisms, Mechanics Based Design of Structures and Machines 31 (2) (2003) 151–179.
[80] M. Zhou, G.I.N. Rozvany, The COC algorithm, part II: topological geometry and generalized shape optimization, Computer Methods in Applied Mechanics and Engineering 89 (1991) 197–224.