Journal of Computational Physics 229 (2010) 5062–5089

Contents lists available at ScienceDirect

Journal of Computational Physics

journal homepage: www.elsevier.com/locate/jcp

Variational piecewise constant level set methods for shape optimization of a two-density drum

Shengfeng Zhu [a,*], Qingbiao Wu [a], Chunxiao Liu [b]

[a] Department of Mathematics, Zhejiang University, Hangzhou 310027 Zhejiang, PR China
[b] Center of Mathematical Sciences, Zhejiang University, Hangzhou 310027 Zhejiang, PR China

ARTICLE INFO

Article history:
Received 31 October 2009
Received in revised form 15 March 2010
Accepted 17 March 2010
Available online 21 March 2010

Keywords:
Shape optimization
Topology optimization
Level set methods
Piecewise constant level set method
Two-phase
Eigenvalue

ABSTRACT

We apply the piecewise constant level set method to a class of eigenvalue related two-phase shape optimization problems. Based on the augmented Lagrangian method and the Lagrange multiplier approach, we propose three effective variational methods for the constrained optimization problem. The corresponding gradient-type algorithms are detailed. The first Uzawa-type algorithm having applied to shape optimization in the literature is proven to be effective for our model, but it lacks stability and accuracy in satisfying the geometry constraint during the iteration. The two other novel algorithms we propose can overcome this limitation and satisfy the geometry constraint very accurately at each iteration. Moreover, they are both highly initial independent and more robust than the first algorithm. Without penalty parameters, the last projection Lagrangian algorithm has less severe restriction on the time step than the first two algorithms. Numerical results for various instances are presented and compared with those obtained by level set methods. The comparisons show effectiveness, efficiency and robustness of our methods. We expect our promising algorithms to be applied to other shape optimization and multiphase problems.

1. Introduction

Optimal shape design is very important and challenging in science and engineering. The typical problem is to find the optimal shape that minimizes or maximizes an objective functional satisfying certain PDE and geometry constraints. The essential difficulty for solving shape optimization problems is that the topology of the optimal shape is unknown *a priori*. One therefore needs to find a mechanism to represent a shape and follow its evolution. Moreover, the changing topology should be handled automatically during the evolution.

One type of such problems arises from structural vibration control and has many engineering design applications [5,42], such as the band structure optimization of photonic crystals [12,16,17]. As a model problem, we consider an acoustic drum head with a fixed bounded domain $\Omega \subset \mathbb{R}^2$ and variable density $\rho(x)$. The resonant frequencies of the drum satisfy the following eigenvalue problem:

$$\begin{cases} -\Delta u(x) = \lambda \rho(x) u(x) & \text{in } \Omega, \\ u(x) = 0 & \text{on } \partial\Omega. \end{cases} \tag{1}$$

Let $S \subset \Omega$ be an unknown domain. Suppose that the density $\rho(x)$ is a piecewise constant function satisfying

* Corresponding author.
E-mail addresses: shengfengzhu@zju.edu.cn (S. Zhu), qbwu@zju.edu.cn (Q. Wu), xxliu198431@126.com (C. Liu).

0021-9991/$ - see front matter © 2010 Elsevier Inc. All rights reserved.
doi:10.1016/j.jcp.2010.03.026

$$\rho(x) = \begin{cases} \rho_1, & \text{if } x \in \Omega \setminus S, \\ \rho_2, & \text{if } x \in S. \end{cases} \tag{2}$$

The mathematical problem we investigate here is to find an optimal distributed parameter $\rho(x)$ that solves the following constrained optimization problem:

$$\min \lambda_1 \text{ or } \max \lambda_1 \text{ or } \max (\lambda_{m+1} - \lambda_m), \quad m = 1, 2, 3, 4, \tag{3}$$

subject to

$$\|S\| = K, \tag{4}$$

where $\|\cdot\|$ denotes the area of a domain and K is some prescribed number.

Osher and Santosa [37] firstly solved the above model problem elegantly by the level set method (LSM). They proposed an effective algorithm by combining the variational LSM [59] and the projection gradient method [43]. They used the Lagrange multiplier technique to convert the constrained optimization problem to an unconstrained one. At each iteration, the Lagrange multiplier was solved using a projection approach based on the linearization of the constraint. When an iteration has violated the constraint by a prescribed tolerance, they found an optimal Lagrange multiplier using Newton's method.

The LSM originally proposed by Osher and Sethian [38] is a versatile tool when dealing with problems required tracing interfaces separating a domain into subregions. The interface is represented implicitly by the zero level set of a Lipschitz continuous level set function (LSF). In the conventional LSM, the Hamilton–Jacobi equation for the LSF is solved to evolve the interface by using a capturing Eulerian approach. Upwind schemes, higher order essentially non-oscillatory (ENO) [39] and weighted ENO (WENO) [25] schemes can be used to solve this equation. During the evolution, often regularity is imposed on the LSF by requiring it to be a signed distance function. The so-called re-initialization process should be performed periodically. There are some effective numerical approaches for re-initialization, such as the fast marching method [44] and the fast sweeping algorithm [55]. LSMs can easily handle certain types of shape and topological changes, such as merging, splitting and developing sharp corners. The methods can therefore be naturally used to solve optimal shape design and topology optimization problems [3,46,57]. For more details about the LSMs and their wide applications, we refer to see [9,11,13,19,35,36,45,50,51] and the references therein.

Motivated by the level set ideas in [37,27], Haber [22] minimized and maximized the first eigenvalue by using a reduced Hessian sequential quadratic programming method combined with multilevel continuation techniques. Strang and Persson [41] solved the eigenvalue problem (1) in an irregular domain using the finite element method on unstructured meshes generated by the LSM [40]. Then they used the gradient descent method of Osher and Santosa [37] to minimize the first and the second eigenvalue. Their method can work with arbitrary domains and resolve the interface by the triangular mesh, but remeshing is needed at each iteration. Recently, Brandman [7] used the LSM to compute the eigenvalues of an elliptic operator defined on a hypersurface, which is represented implicitly as the zero level set of a LSF.

In classical shape sensitivity analysis for shape optimization problems, shape derivatives that measure the sensitivity of boundary or interface perturbations are derived to obtain the shape gradient of the objective functional. For detailed theoretical analysis and applications of shape sensitivity analysis, we refer to see [48] and the references therein. After calculation of the shape gradient, gradient-type algorithms are used to decrease the objective functional and stop if the shape gradient vanishes. But such methods tend to fall into local minima and are generally implemented under the Lagrangian framework which requires remeshing at each iteration. The homogenization method [1] can overcome the two drawbacks, but it is mainly restricted to linear elasticity and gives optimal shapes that are composite. Penalization methods are needed to project the composite shape on a classical two-phase design.

The combination of LSMs with the shape sensitivity analysis framework has become a standard tool for solving a variety of shape optimization and inverse problems in engineering [2,3,8,9,15,57,60]. Fast Newton-type shape optimization methods were used for level set formulations in [24]. However, as pointed out in [2,3,10,20,23,56], the conventional LSM based on shape sensitivity analysis cannot create new holes automatically during the evolution which may get stuck at shapes with fewer holes than the optimal geometry. Therefore, the initial shape guess generally contains many holes. To eliminate this weakness, topology derivatives were incorporated into shape derivatives based LSMs for inverse obstacle problems [10], structure optimization [4] and shape optimization problems [20]. The topology derivative introduced firstly by Sokołowski and Żochowski [47] measures the influence of creating small holes centered at a certain point in the domain. Motivated by the idea in [10], He et al. [23] combined shape derivatives with topological derivatives in LSMs to maximize band gaps and presented an algorithm more efficient and flexible in topology changing than the original LSM based on the shape derivatives. For optimizing spectral gaps of the drum, they improved the numerical results in [37].

As recent variants of the standard LSMs, piecewise constant level set methods (PCLSMs) were proposed by Tai et al. [31,52] for image segmentation and elliptic inverse problems [53]. Similar ideas can be found in [21,30,33,49]. The systematic and general framework of the PCLSM was presented in [29]. Different from standard LSMs, the PCLSM can identify subregions using one discontinuous piecewise constant level set function (PCLSF) which can only take piecewise constant values at convergence. One has to employ N LSFs to represent up to 2^N subregions in the LSM, while PCLSM can use one LSF to distinguish multiple regions. The LSM propagates the interface by defining speed only on the interface, which makes it generally could not create small holes at the places far away from the interface. The PCLSM determines the interface by forcing the value of the LSF at each mesh point to be one of the piecewise constant values. Therefore, solving shape and topology

problems by the PCLSM can create holes during the evolution of the LSF without topological derivatives. One can therefore develop more complicated shapes than the conventional LSM. Another merit is that one does not need to solve the Hamilton–Jacobi equation any more and the periodic re-initialization process is eliminated. Furthermore, iterative algorithms having little dependence on the initial guess can be designed. Recently, based on the PCLSM formulation, the augmented Lagrangian method [6,34,54] was employed to solve the structural topology optimization problems [58] and optimal shape design of multi-material piezoelectric actuators with in-plane motion [32].

In this paper, we devise algorithms for solving shape optimization problems under the piecewise constant level set framework. By introducing a LSF, the original model problem is formulated as an optimization problem with two constraints. The first constraint is the original area constraint and the second one is added to force the LSF to become a PCLSF. By the augmented Lagrangian and Lagrange multiplier techniques for constrained optimization problems, we propose three types of variational algorithms. The first one is a Uzawa-type algorithm based on the augmented Lagrangian method for the two constraints. Uzawa-type algorithms under the piecewise constant level set frame have proven to be effective for shape and topology optimization [32,58]. We apply such an algorithm to our model problem and illustrate its effectiveness in numerical simulation. But we observe that the area constraint cannot be satisfied well even after a number of iterations. To overcome this flaw, we then present the next two methods, one of which is a hybrid algorithm by coupling the Lagrange multiplier technique for the area constraint with the augmented Lagrangian method for the other constraint. The last algorithm without penalty parameters suffers from less severe restriction on the time step size compared with the first two algorithms. In this algorithm, we employ the same method as the second algorithm in dealing with the area constraint. But we use a projection Lagrangian method similar as in [30] to construct an iterative scheme for the second Lagrange multiplier. An acceleration technique is given to speed up and improve the algorithm.

The rest of the paper is organized as follows. In Section 2, we present the basic formulations of the PCLSM. In Section 3, the PCLSM is applied to the eigenvalue shape design problem. Based on the augmented Lagrangian method and the Lagrange multiplier approach, we present three algorithms. Numerical results for the model problem are presented in Section 4. Finally, we conclude the paper in the last section and outline our future work.

2. Piecewise constant level set method

Assume that the domain Ω is partitioned into n subdomains $\{\Omega_i\}_{i=1}^n$ such that

$$\overline{\Omega} = \bigcup_{i=1}^{n} \Omega_i \cup \Gamma, \tag{5}$$

where Γ is the union of the boundaries of the subregions. In order to identify the subregions, we define a n-phase PCLSF $\phi\,\Omega \mapsto \mathbb{R}$ as follows:

$$\phi = i \quad \text{in } \Omega_i, \;\; i = 1, 2, \ldots, n. \tag{6}$$

Then the characteristic functions of the subdomains are represented as

$$\chi_i = \frac{1}{\alpha_i} \prod_{j=1, j\neq i}^{n} (\phi - j) \quad \text{with } \alpha_i = \prod_{k=1, k\neq i}^{n} (i - k). \tag{7}$$

By the properties of the characteristic functions, we can calculate the area inside Ω_i and the length of the boundary of Ω_i, respectively, by

$$\|\Omega_i\| = \int_\Omega \chi_i \mathrm{d}x \quad \text{and} \quad |\partial\Omega_i| = \int_\Omega |\nabla \chi_i| \,\mathrm{d}x. \tag{8}$$

It should be pointed out that we avoid the non-differentiability of the Heaviside and Delta functions in the standard LSMs. It is therefore not necessary to use the smoothed and regularized counterparts for them.

We denote

$$H(\phi) = (\phi - 1)(\phi - 2)\ldots(\phi - n) = \prod_{i=1}^{n} (\phi - i). \tag{9}$$

Then (6) implies that

$$H(\phi) = 0 \quad \text{in } \Omega. \tag{10}$$

The piecewise constant constraint (10) guarantees that there is no vacuum and overlap between different subregions. Any piecewise constant function $\rho(x)$ with $\rho = \rho_i$ in Ω_i can be expressed as

$$\rho(x) = \sum_{i=1}^{n} \rho_i \chi_i(\phi). \tag{11}$$

Now we can apply the two-phase PCLSM formulation to our two-density model problem (3). Let $\Omega_1 = S$ and $\Omega_2 = \Omega \setminus S$. Here $\phi = 1$ and $\phi = 2$ correspond to Ω_1 and Ω_2, respectively. Then $\|S\| = \int_\Omega (2-\phi)\mathrm{d}x$. The density ρ can be represented as

$$\rho(\phi) = \rho_1\chi_1 + \rho_2\chi_2 = \rho_1(\phi - 1) + \rho_2(2-\phi). \tag{12}$$

Let $F(\phi) = \lambda_1$ or $-\lambda_1$ or $\lambda_m - \lambda_{m+1}$ for $m = 1, 2, 3, 4$. Then the generic eigenvalue problem (3) and (4) can be formulated as

$$\min_{\phi} F(\phi) \quad \text{subject to } G(\phi) = 0 \text{ and } H(\phi) = 0, \tag{13}$$

where

$$G(\phi) = \int_\Omega (2-\phi)\mathrm{d}x - K, \tag{14}$$

$$H(\phi) = (\phi - 1)(\phi - 2). \tag{15}$$

We let for example $F(\phi) = \lambda_1$ to show how to calculate the gradient of $F(\phi)$. Suppose that the first eigenpair (u_1, λ_1) solves (1), i.e., it satisfies

$$-\Delta u_1 = \lambda_1 \rho u_1 \quad \text{in } \Omega, \tag{16}$$

$$u_1 = 0 \quad \text{on } \partial\Omega. \tag{17}$$

Differentiating w.r.t. ρ in a direction $v \in C_0^\infty(\Omega)$ on both sides of (16) leads to

$$-\Delta\left(\frac{\partial u_1}{\partial \rho}\cdot v\right) - \lambda_1\rho\left(\frac{\partial u_1}{\partial \rho}\cdot v\right) = \rho u_1\left(\frac{\partial \lambda_1}{\partial \rho}\cdot v\right) + \lambda_1 v u_1. \tag{18}$$

Multiplying both sides by u_1, integrating and using Green's theorem, we have

$$\begin{aligned}\int_\Omega \left[\rho u_1\left(\frac{\partial \lambda_1}{\partial \rho}\cdot v\right) + \lambda_1 v u_1\right] u_1 \mathrm{d}x &= \int_\Omega \left[-\Delta\left(\frac{\partial u_1}{\partial \rho}\cdot v\right) - \lambda_1\rho\left(\frac{\partial u_1}{\partial \rho}\cdot v\right)\right] u_1 \,\mathrm{d}x \\ &= \int_\Omega \left[\nabla u_1 \cdot \nabla\left(\frac{\partial u_1}{\partial \rho}\cdot v\right) - \lambda_1\rho u_1\left(\frac{\partial u_1}{\partial \rho}\cdot v\right)\right]\mathrm{d}x \\ &= \int_\Omega (-\Delta u_1 - \lambda_1\rho u_1)\left(\frac{\partial u_1}{\partial \rho}\cdot v\right)\mathrm{d}x = 0.\end{aligned} \tag{19}$$

Thus, the directional derivative

$$\frac{\partial \lambda_1}{\partial \rho}\cdot v = -\frac{\lambda_1 \int_\Omega v u_1^2 \mathrm{d}x}{\int_\Omega \rho u_1^2 \mathrm{d}x}, \tag{20}$$

which implies that the Gâteaux differential

$$\frac{\partial \lambda_1}{\partial \rho} = -\frac{\lambda_1 u_1^2}{\int_\Omega \rho u_1^2 \mathrm{d}x}. \tag{21}$$

By the chain rule, we have

$$\frac{\partial \lambda_1}{\partial \phi} = \frac{\partial \lambda_1}{\partial \rho}\frac{\partial \rho}{\partial \phi} = \frac{(\rho_2 - \rho_1)\lambda_1 u_1^2}{\int_\Omega [\rho_1(\phi - 1) + \rho_2(2-\phi)] u_1^2 \mathrm{d}x}. \tag{22}$$

3. Algorithms

For the constrained optimization problem (13), we need to transform it to an unconstrained one. Different treatments for the two constraints lead to different formulations. Generally, there are three essential strategies: augmented Lagrangian method, Lagrange multiplier method and penalty approach.

In this section, we will present three iterative algorithms: Uzawa-type algorithm, hybrid algorithm and projection Lagrangian algorithm. They all rely on gradient descent flows by introducing an artificial time variable t. In the following, the upperscript k denotes the kth iteration.

3.1. Augmented Lagrangian algorithm

The augmented Lagrangian method has been studied well in [6,34] and applied widely for various problems (see e.g. [13,14,26,54]). Recently, this classic method was used successfully under the piecewise constant level set framework for solving shape and topology related optimization problems [28–32,58].

First, we use the augmented Lagrangian method to convert the problem (13) to an unconstrained one

$$\min_{\phi}\max_{l_1,l_2}\mathcal{L}(\phi,l_1,l_2)=F(\phi)+l_1G(\phi)+\frac{1}{2\mu_1}G^2(\phi)+\int_\Omega l_2H(\phi)\mathrm{d}x+\frac{1}{2\mu_2}\int_\Omega H^2(\phi)\mathrm{d}x, \tag{23}$$

where $l_1\in\mathbb{R}$ and $l_2\in L^2(\Omega)$ are Lagrange multipliers and μ_1 and $\mu_2>0$ are penalty parameters. A minimizer of $F(\phi)$ corresponds to a saddle point of the augmented Lagrangian functional $\mathcal{L}$. The system of necessary conditions for a saddle point of $\mathcal{L}$ is

$$\frac{\partial\mathcal{L}}{\partial\phi}=0,\quad \frac{\partial\mathcal{L}}{\partial l_1}=0\quad\text{and}\quad \frac{\partial\mathcal{L}}{\partial l_2}=0, \tag{24}$$

where

$$\frac{\partial\mathcal{L}}{\partial\phi}=\frac{\partial F}{\partial\phi}+l_1\frac{\partial G}{\partial\phi}+\frac{1}{\mu_1}G\frac{\partial G}{\partial\phi}+l_2\frac{\partial H}{\partial\phi}+\frac{1}{\mu_2}H\frac{\partial H}{\partial\phi}, \tag{25}$$

$$\frac{\partial\mathcal{L}}{\partial l_1}=G(\phi)=\int_\Omega(2-\phi)\mathrm{d}x-K, \tag{26}$$

$$\frac{\partial\mathcal{L}}{\partial l_2}=H(\phi)=(\phi-1)(\phi-2). \tag{27}$$

More precisely, for (25), we have

$$\begin{aligned}\frac{\partial\mathcal{L}}{\partial\phi}&=\frac{\partial\lambda_1}{\partial\rho}\frac{\partial\rho}{\partial\phi}+l_1\frac{\partial G}{\partial\phi}+\frac{1}{\mu_1}G\frac{\partial G}{\partial\phi}+l_2\frac{\partial H}{\partial\phi}+\frac{1}{\mu_2}H\frac{\partial H}{\partial\phi}\\&=\frac{(\rho_2-\rho_1)\lambda_1u_1^2}{\int_\Omega[\rho_1(\phi-1)+\rho_2(2-\phi)]u_1^2\mathrm{d}x}-l_1-\frac{1}{\mu_1}\left[\int_\Omega(2-\phi)\mathrm{d}x-K\right]+l_2(2\phi-3)+\frac{1}{\mu_2}(\phi-1)(\phi-2)(2\phi-3).\end{aligned} \tag{28}$$

We will update ϕ, l_1 and l_2 alternatively. To minimize $\mathcal{L}$ w.r.t. ϕ, we should introduce an artificial time variable t and solve the PDE

$$\begin{cases}\frac{\partial\phi}{\partial t}=-\frac{\partial\mathcal{L}}{\partial\phi} & \text{in } \Omega\times\mathbb{R}^+,\\ \phi(x,0)=\phi_0(x) & \text{in } \Omega\end{cases} \tag{29}$$

to the steady state $\partial\phi/\partial t=0$, which implies that $\partial\mathcal{L}/\partial\phi=0$. We discretize (29) by a forward Euler scheme

$$\phi^{k+1}=\phi^k-\Delta t^k\frac{\partial\mathcal{L}}{\partial\phi}(\phi^k,l_1^k,l_2^k), \tag{30}$$

where Δt^k is the time step. The updating scheme (30) for ϕ is essentially the gradient descent method for minimization of $\mathcal{L}$ w.r.t. ϕ. We can find the optimal Δt^k by the Armijo–Goldstein search method or the Wolfe–Powell rule in each iteration. Compared with setting a small fixed time step by experience, a line search based gradient descent method can reduce the number of the whole iterations. But the computational effort at each iterate will increase. Alternatively, considering the Courant–Friedrichs–Lewy (CFL) condition for stability, we may set

$$\Delta t^k=\sigma h/\max_{x\in\Omega}\left|\frac{\partial\mathcal{L}}{\partial\phi}\left(\phi^k(x),l_1^k,l_2^k\right)\right|, \tag{31}$$

where h is the mesh size and the positive constant σ can be chosen by the trial and error method. This way of setting the time step for gradient evolution was also used in [10].

The Lagrange multipliers can be updated by the Uzawa-type scheme [29,31,32,58]:

$$l_1^{k+1}=l_1^k+\frac{1}{\mu_1^k}\left[\int_\Omega\left(2-\phi^{k+1}\right)\mathrm{d}x-K\right], \tag{32}$$

$$l_2^{k+1}=l_2^k+\frac{1}{\mu_2^k}(\phi^{k+1}-1)(\phi^{k+1}-2). \tag{33}$$

For the augmented Lagrangian method, due to the Lagrange multipliers, the constraints can be satisfied even if we use fixed penalization parameters during the iteration [30]. In practice, however, better convergence can be obtained if we decrease gradually the values of the penalization parameters. We set

$$\mu_i^{k+1}=\theta_i\mu_i^k,\quad \theta_i\in(0,1),\ \ i=1,2, \tag{34}$$

where the decreasing factors θ_1 and θ_2 are constants required to be chosen properly.

Now we are ready to present the following Uzawa-type algorithm to obtain a saddle point of (23).

Algorithm 1. *Augmented Lagrangian algorithm*

Initialize ϕ^0, l_1^0, l_2^0, μ_1^0 and μ_2^0. For $k=0,1,2,\ldots$,

Step 1. Use (12) to calculate $\rho^k = \rho_1(\phi^k - 1) + \rho_2(2 - \phi^k)$. Solve (16) and (17) with ρ replaced by ρ^k and obtain $(\lambda_1^k,\ u_1^k)$.
Step 2. Set Δt^k by (31) and use the scheme (30) to update the LSF.
Step 3. Update the Lagrange multipliers and penalty parameters by (32)–(34).
Step 4. If not converged: Set $k = k + 1$ and go to step 1.

The iterative procedure can be terminated when it satisfies

$$\left\|\frac{\partial \mathcal{L}}{\partial \phi}\left(\phi^{k+1}, l_1^{k+1}, l_2^{k+1}\right)\right\|_{L^2(\Omega)} \leqslant \eta\left\|\frac{\partial \mathcal{L}}{\partial \phi}(\phi^k, l_1^k, l_2^k)\right\|_{L^2(\Omega)}, \quad \eta \in (0,1) \tag{35}$$

or reaches some prescribed iteration step.

Remark 1. The time step Δt^k set by (31) is influenced by μ_1 and μ_2. A small μ_1^k or μ_2^k leads to a small Δt^k. Implicit or semi-implicit schemes can be applied to make larger time steps in solving the PDE (30), but it is not the focus of this paper.

Remark 2. Besides the augmented Lagrangian method, the piecewise constant constraint can be treated by the penalty approach. The penalty method is simple and behaves more stable, but it is hard to satisfy the constraint exactly unless the penalty parameter μ_2 is set to be small enough. However, the small value of the penalty parameter may cause instability. With the augmented Lagrangian method for this constraint, there is no need to set the penalty parameter very small. Furthermore, the effective Uzawa algorithm can be applied under the augmented Lagrangian frame.

Remark 3. The manner of setting the penalization parameters μ_1 and μ_2 and has a great impact on the convergence process and the topology of the computed optimal shape. We can set them to be constant during the iterations. For better convergence behavior, however, we choose large values for μ_1^0 and μ_2^0 at the beginning and decrease them gradually as (34) or we can start to update them after some steps. If μ_1^0 and μ_2^0 are too small, the minimization focuses on the penalization of violations of the constraints. Then ϕ may converge to 1 and 2 before the algorithm can determine the optimal topology. It is therefore very likely to get stuck at local minima. On the other hand, if μ_1 and μ_2 are too large, penalization on the constraints is not enough and the iteration is not stable. The value of ϕ can deviate from 1 and 2 greatly.

3.2. Hybrid algorithm

By the Uzawa Algorithm, the area constraint can be satisfied approximately after rather many iterations. To satisfy the constraint more stably and accurately, we use the Lagrange multiplier method to treat this constraint. The optimal Lagrange multiplier is obtained at each iteration by solving a one-dimensional optimization problem. By mixing the Lagrange multiplier method for the area constraint and the augmented Lagrangian method for the piecewise constant constraint, we devise a more effective and accurate hybrid algorithm than Algorithm 1 for the model problem.

We omit the first penalization term in (23) and obtain a hybrid Lagrangian functional

$$\widehat{\mathcal{L}}(\phi, l_1, l_2) = F(\phi) + l_1 G(\phi) + \int_\Omega l_2 H(\phi)\mathrm{d}x + \frac{1}{2\mu_2}\int_\Omega H^2(\phi)\mathrm{d}x. \tag{36}$$

At a saddle point of $\widehat{\mathcal{L}}$, we have

$$\begin{aligned}\frac{\partial \widehat{\mathcal{L}}}{\partial \phi} &= \frac{\partial F}{\partial \phi} + l_1\frac{\partial G}{\partial \phi} + l_2\frac{\partial H}{\partial \phi} + \frac{1}{\mu_2}H\frac{\partial H}{\partial \phi} \\ &= \frac{(\rho_2 - \rho_1)\lambda_1 u_1^2}{\int_\Omega[\rho_1(\phi-1) + \rho_2(2-\phi)]u_1^2\mathrm{d}x} - l_1 + l_2(2\phi - 3) + \frac{1}{\mu_2}(\phi-1)(\phi-2)(2\phi-3) = 0, \end{aligned} \tag{37}$$

$$\frac{\partial \widehat{\mathcal{L}}}{\partial l_1} = \frac{\partial \mathcal{L}}{\partial l_1} = 0, \quad \frac{\partial \widehat{\mathcal{L}}}{\partial l_2} = \frac{\partial \mathcal{L}}{\partial l_2} = 0. \tag{38}$$

Osher and Santosa [37] calculated l_1 from the linearized equation of constraint $G(\phi) = 0$. Then they used Newton's method to update l_1 when an iteration has violated the area constraint by a specified tolerance. This approach can keep the area of S conservative every several steps. Here we apply their projection method and differentiate the constraint equation w.r.t. ϕ in the direction $\delta\phi$:

$$\left\langle \frac{\partial G}{\partial \phi}, \delta\phi \right\rangle = 0, \tag{39}$$

where $\delta\phi$ is a variation in ϕ and $\langle , \rangle$ denotes the L^2 inner product. Choose $\delta\phi = \partial\phi/\partial t$ and we have

$$\left\langle \frac{\partial G}{\partial \phi}, \frac{\partial \phi}{\partial t} \right\rangle = \left\langle \frac{\partial G}{\partial \phi}, -\frac{\partial \widehat{\mathcal{L}}}{\partial \phi} \right\rangle = \int_\Omega \frac{\partial \widehat{\mathcal{L}}}{\partial \phi}\mathrm{d}x = 0. \tag{40}$$

Thus, from (37) and (40), we obtain

$$l_1 = \frac{1}{\|\Omega\|}\int_\Omega \left\{ \frac{(\rho_2-\rho_1)\lambda_1 u_1^2}{\int_\Omega[\rho_1(\phi-1)+\rho_2(2-\phi)]u_1^2 dx} + l_2(2\phi-3) + \frac{1}{\mu_2}(\phi-1)(\phi-2)(2\phi-3) \right\} dx \triangleq \Xi(\phi, l_2). \tag{41}$$

If an iteration has violated the area constraint too far away from the constraint set, we solve for l_1 via Newton's method in (39), i.e., update l_1 by

$$l_1 \leftarrow l_1 - \left[\frac{\partial G(\phi+\delta\phi(x,l_1,l_2))}{\partial l_1}\right]^{-1} G(\phi+\delta\phi(x,l_1,l_2)) \tag{42}$$

for a few steps.

Alternatively, we adopt another way to seek the optimal l_1 by

$$l_1^{k+1} \leftarrow \arg\min_{l_1} |G(\hat{\phi}^{k+1})|^2, \tag{43}$$

where

$$\hat{\phi}^{k+1} = \phi^k - \Delta t^k \frac{\partial \widehat{\mathcal{L}}}{\partial \phi}(\phi^k, l_1, l_2^k). \tag{44}$$

For the updating of ϕ, l_2 and μ_2, we follow the similar method as in the above subsection. We only need to replace $\partial\mathcal{L}/\partial\phi$ by $\partial\widehat{\mathcal{L}}/\partial\phi$ in (29)–(31). More precisely,

$$\phi^{k+1} = \phi^k - \Delta t^k \frac{\partial \widehat{\mathcal{L}}}{\partial \phi}(\phi^k, l_1^{k+1}, l_2^k) \quad \text{with } \Delta t^k = \sigma h / \max_{x\in\Omega} \left| \frac{\partial \widehat{\mathcal{L}}}{\partial \phi}\left(\phi^k(x), l_1^{k+1}, l_2^k\right)\right|. \tag{45}$$

Algorithm 2. *Hybrid algorithm*

Initialize ϕ^0, l_1^0, l_2^0 and μ_2^0. For $k = 0, 1, 2, \ldots$,

Step 1. Use (12) to calculate $\rho^k = \rho_1(\phi^k - 1) + \rho_2(2 - \phi^k)$. Solve (16) and (17) with ρ replaced by ρ^k and obtain (λ_1^k, u_1^k).

Step 2. Compute the Lagrange multiplier $l_1^{k+1} = \Xi(\phi, l_2^k)$ via (41).

When $|G(\hat{\phi}^{k+1}t)| > \varepsilon$ for a prescribed tolerance ε, we solve for l_1^{k+1} by (43).

Step 3. Use (45) to update ϕ^k.

Step 4. Update l_2 and μ_2 by (33) and (34), respectively.

Step 5. If not converged: Set $k = k + 1$ and go to step 1.

There is rarely theoretical convergence analysis for such a hybrid scheme in the literature and monographs on constrained optimization. However, we shall illustrate its effectiveness and robustness in numerical experiments.

Remark 4. To put an iteration back onto the feasible set, the manner for updating the first Lagrange multiplier in Algorithm 2 is different from Newton's method applied in [37]. We seek the optimal Lagrange multiplier by solving a one-dimensional optimization problem, while the algorithm in [37] updated it through solving a nonlinear equation for l_1. Although they are exactly the same if we use the Newton method in (43), we can apply other optimization methods, such as the line search based steepest descent approach or Quasi-Newton method, to achieve this.

3.3. Projection Lagrangian algorithm

In this section, we propose another so-called projection Lagrangian method based on the general Lagrange multiplier method for both constraints. One advantage of this algorithm compared with Algorithms 1 and 2 is that no penalty parameter is used. With no need to adjust penalty parameters and their decreasing rates, Algorithm 3 is more robust than the two other algorithms. Another virtue worth of mention is that the restrictions on the time step size in Algorithms 1 and 2 are relaxed as shown in numerical simulation. This virtue of Algorithm 3 makes it determine the optimal topology earlier than the first two algorithms.

The projection Lagrangian algorithm [18] was applied to image segmentation under a framework of the binary level set method in [30]. Motivated by this idea, here we employ it under the piecewise constant level set frame to deal with the piecewise constant constraint. For the area constraint, we again use the projected Lagrange multiplier method in Algorithm 2.

Omitting the penalty terms in (23), we define the Lagrangian functional for (13) as

$$L(\phi, l_1, l_2) = F(\phi) + l_1 G(\phi) + \int_\Omega l_2 H(\phi) dx. \tag{46}$$

Similarly, the saddle point of L requires

$$\frac{\partial L}{\partial \phi} = \frac{\partial \lambda_1}{\partial \rho}\frac{\partial \rho}{\partial \phi} + l_1\frac{\partial G}{\partial \phi} + l_2\frac{\partial H}{\partial \phi} \tag{47}$$

$$= \frac{(\rho_2-\rho_1)\lambda_1 u_1^2}{\int_\Omega[\rho_1(\phi-1)+\rho_2(2-\phi)]u_1^2 dx} - l_1 + l_2(2\phi-3) = 0, \tag{48}$$

$$\frac{\partial L}{\partial l_1} = G(\phi) = 0, \quad \frac{\partial L}{\partial l_2} = H(\phi) = 0. \tag{49}$$

We use the same technique as (41) and (43) to find the optimal Lagrange multiplier l_1 at each iteration. We have

$$l_1 = \frac{1}{\|\Omega\|}\int_\Omega \left\{ \frac{(\rho_2-\rho_1)\lambda_1 u_1^2}{\int_\Omega[\rho_1(\phi-1)+\rho_2(2-\phi)]u_1^2 dx} + l_2(2\phi-3)\right\} dx \triangleq \Gamma(\phi, l_2). \tag{50}$$

If the area constraint cannot be satisfied well with l_1 computed by (50), we solve for l_1 by

$$l_1^{k+1} \leftarrow \arg\min_{l_1} \left| G\left(\tilde{\phi}^{k+1}\right)\right|^2, \tag{51}$$

where

$$\tilde{\phi}^{k+1} = \phi^k - \Delta t^k \frac{\partial L}{\partial \phi}(\phi^k, l_1, l_2^k). \tag{52}$$

We use the similar gradient descent method as (30) to update ϕ by

$$\phi^{k+1} = \phi^k - \Delta t^k \frac{\partial L}{\partial \phi}(\phi^k, l_1^{k+1}, l_2^k), \tag{53}$$

where $\Delta t^k = \sigma h / \max_{x\in\Omega}|\frac{\partial L}{\partial \phi}\left(\phi^k(x), l_1^{k+1}, l_2^k\right)|$.

Now we discuss how to construct an updating scheme for l_2 by (48). Multiplying this equation by $2\phi-3$ and using the constraint $(\phi-1)(\phi-2)=0$ in (49), we get the explicit expression of l_2

$$l_2 = (2\phi-3)\left\{\frac{(\rho_1-\rho_2)\lambda_1 u_1^2}{\int_\Omega[\rho_1(\phi-1)+\rho_2(2-\phi)]u_1^2 dx} + l_1\right\} \triangleq \Psi(\phi, l_1). \tag{54}$$

The updating scheme (54) is in essence a projection Lagrangian algorithm of Uzawa type [18,30].

Algorithm 3. *Projection Lagrangian algorithm*

Initialize ϕ^0, l_1^0 and l_2^0. For $k = 0, 1, 2, \ldots$,

Step 1. Use (12) to compute $\rho^k = \rho_1(2-\phi^k) + \rho_2(\phi^k - 1)$. Solve (16) and (17) with ρ replaced by ρ^k and obtain (λ_1^k, u_1^k).

Step 2. Update l_1 by $l_1^{k+1} = \Gamma(\phi^k, l_2^k)$.

When $|G(\tilde{\phi}^{k+1})| > \varepsilon$, use (51) to get l_1^{k+1}.

Step 3. Update the LSF by (53).

Step 4. Update l_2 by $l_2^{k+1} = \Psi(\phi^{k+1}, l_1^{k+1})$.

Step 5. If not converged: Set $k = k+1$ and go to step 1.

Remark 5. Based on the binary level set framework, the projection Lagrangian algorithm proposed in [30] is similar to our Algorithm 3 in updating the Lagrange multiplier l_2. The main difference is that we have to treat the area constraint in our problem.

4. Numerical experiments

In this part, we will test the proposed algorithms on the eigenvalue shape optimization problem. We solve all examples on a domain $\Omega = [0,1]\times[0,1.5]$ with the mesh size $h = 1/40$ and use the five-point difference scheme to discretize the Laplacian operator. All numerical experiments are performed in MATLAB on a PC with Intel Core 2 Duo 2.33 GHz processor and 2 GB memory. One iteration usually costs 0.07–0.08 s and every numerical example takes less than 1 min. Our algorithms are more efficient than LSMs mainly because we do not need to implement the periodical re-initialization process. The subroutine `eigs` is called for solving the generalized eigenvalue problem at each iteration. We use `contourf` to plot the two-density distribution. In all the following figures for the density distribution, white corresponds to $\rho_2 = 2$.

The constraint $H(\phi) = 0$ can be replaced by $H^\alpha(\phi) = 0$ for $\alpha > 1$. As described in [58], the smaller the α value, the faster the convergence speed. In our computations, the case $\alpha = 1$ is used. Let $\|H(\phi)\|_2$ denote the computed L^2 norm of $H(\phi)$. This quantity measures the constraint violations.

The area constraint is equivalent to a mass constraint if we calculate the mass of the drum by

$$M := \rho_1(\|\Omega\| - \|S\|) + \rho_2\|S\| = \rho_1\|\Omega\| + (\rho_2 - \rho_1)\int_\Omega (2-\phi)\mathrm{d}x. \tag{55}$$

We set $K = \|\Omega\|/2 = 0.75$, i.e., the mass of the drum is 2.25. We can use M to numerically measure if the geometry constraint is satisfied during the iteration. The mass quantity M in our first algorithm usually oscillates around 2.25 in the first few iterations. After convergence, the mass constraint satisfaction is not very satisfactory even if an initial LSF satisfying exactly this constraint is chosen. Algorithms 2 and 3 are numerically mass conservative, namely, the computed quantity M can approximate to 2.25 very accurately at each iteration. The final mass value is rather close to 2.25 even starting from a poor initial guess. However, the density represented by (12) is composite when ϕ has not converged to 1 and 2. Therefore, the following figures for the evolution of the two-phase density plotted by `controuf` visually show that the constraint is not satisfied until convergence. The tolerance ε is set to be 10^{-4}.

4.1. Uzawa-type algorithm

Let us first test the performance of the Algorithm 1 for our shape design problem. The algorithm starts with $\mu_1^0 = 10,\ \mu_2^0 = 50,\ l_1^0 = 0,\ l_2^0 = 0$ and $\phi^0 = 1.5$. We choose $\theta_1 = \theta_2 = 0.95$ and $\sigma = 0.5$. Consider the two cases: $\min\lambda_1$ and $\max\lambda_1$ as illustrations for effectiveness and limitations of the algorithm. Evolutions of the LSF and densities for minimizing λ_1 are shown in Fig. 1. See Fig. 2 for the evolution of densities for maximizing λ_1. In Fig. 3, we show the convergence history of λ_1 for the two experiments. After 400 iterations, $\lambda_1 = 7.3218$ and 13.2742, respectively. Compared the two values respectively with another two optimal eigenvalues 7.37 and 13.45 obtained using LSM [37], our first value is slightly better, while the second one is a little worse.

For both examples, Fig. 4 demonstrates that the algorithm oscillates first and begins to satisfy the mass constraint approximately after about 200 iterations. After 400 iterations, the error $M - 2.25$ is about -9.30×10^{-3} and 6.27×10^{-3}, respectively. The two values seem to be tolerable, but the error can hardly be reduced further by Algorithm 1. In order to improve both the accuracy and stability of the mass constraint satisfaction, we use the Lagrange multiplier method in the next two algorithms to deal with this constraint.

4.2. Hybrid algorithm

Comparing with Algorithm 1, the hybrid algorithm requires to solve a one-dimensional optimization problem when the absolute error between the computed mass and 2.25 is larger than the tolerance ε. However, this additional computational effort is nearly negligible from numerical experience. In this part, we set $\sigma = 0.4,\ \mu_2^0 = 100$ and $\theta_2 = 0.90$ for the following numerical examples.

4.2.1. Case 1: $\min\lambda_1$

To illustrate the hybrid Algorithm 2 is almost independent of the initial guess, we display evolutions of the LSF and/or densities from four different initial density distributions, respectively, in Figs. 5–8. In Fig. 6, we obtained satisfactory results even if we choose a random initial guess. Moreover, our algorithm can deal with the area constraint accurately. In Fig. 9, we demonstrate numerically mass preserving of the algorithm although this cannot be seen from the density distributions in Figs. 5–8. The convergence history of $M,\ \|H(\phi)\|_2$ and λ_1 for the four initial designs are presented in Figs. 9–11, respectively. From the curves in these three figures, we can see the algorithm is stable and robust. Table 1 demonstrates minor differences among the results of the four evolutions. After 400 iterations, the constraint $H(\phi) \approx 0.05$, i.e., ϕ converges approximately to 1 or 2 at every grid point. The final value of λ_1 approximates to 7.3 for each design.

4.2.2. Case 2: $\max\lambda_1$

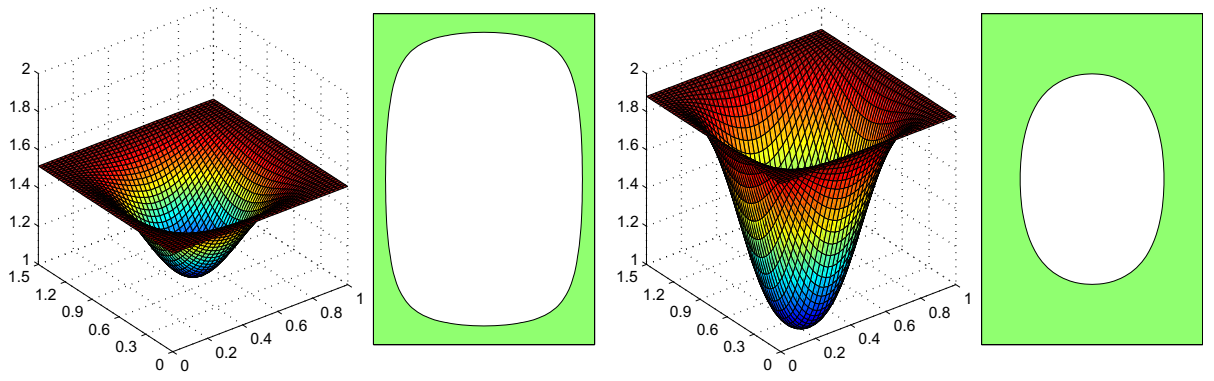
Fig. 12 shows evolutions of the LSF and the densities from the same initial design as in Fig. 5(a). In Fig. 13, the value of λ_1 increases monotonically after about 50 iterations. As shown in Fig. 14, there are only some minor oscillations for M in the first 150 iterations. The initial value for $\|H(\phi)\|_2$ is zero. The value of $\|H(\phi)\|_2$ increases first but deceases after about 100 iterations.

4.2.3. Case 3: $\max(\lambda_2 - \lambda_1)$

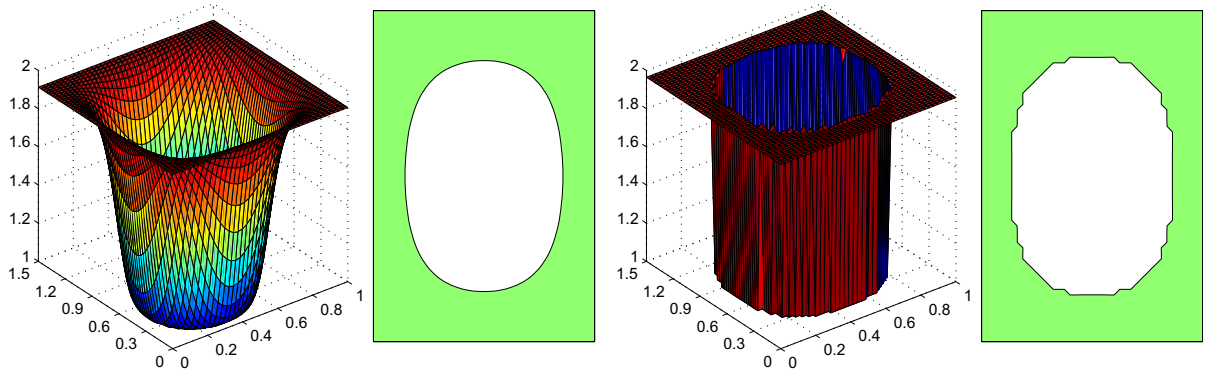
This example demonstrates the process of maximizing the gap between λ_2 and λ_1. Fig. 15 shows evolutions of the LSF and the densities. See Fig. 16 for corresponding convergence history of $\lambda_1,\ \lambda_2$ and $\lambda_2 - \lambda_1$. The quantities M and $\|H(\phi)\|_2$ are plotted in Fig. 17.

In Table 2, the comparison between Algorithm 2 and LSM [37] shows that our algorithm leads to a smaller optimum than LSM for Case 1, while the LSM performs better for Case 2 and Case 3.

Remark 6. The algorithm can finally develop sharp discontinuities at the interface due to the penalization effect on the piecewise constant constraint. But the converged ϕ deviates slightly from 1 and 2 as shown in the evolution figures above.



(a) Step 40 (b) Step 120



(c) Step 120 (d) Step 400

Fig. 1. Minimization of λ_1 by Algorithm 1: evolutions of the LSF and densities from initial design $\phi^0 = 1.5$.

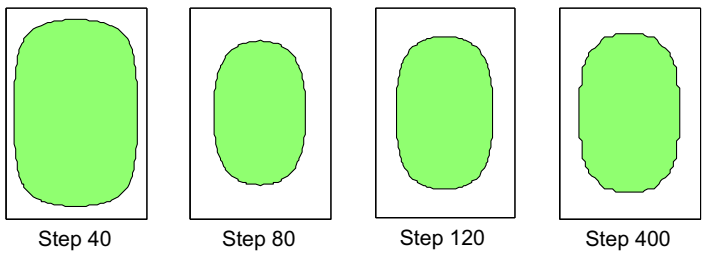


Fig. 2. Maximization of λ_1 by Algorithm 1: evolution of the densities from initial design $\phi^0 = 1.5$.

4.3. Projection Lagrangian algorithm

As will shown below, the CFL stability condition in this algorithm allows a relatively larger time step than that in Algorithms 1 and 2 from numerical experience.

The LSF ϕ in the region far from the interface converges quickly to 1 and 2. But it takes rather many iterations for ϕ in the region near the interface to reach them, i.e., form the discontinuity. To speed up the projection Lagrangian method, we introduce the function

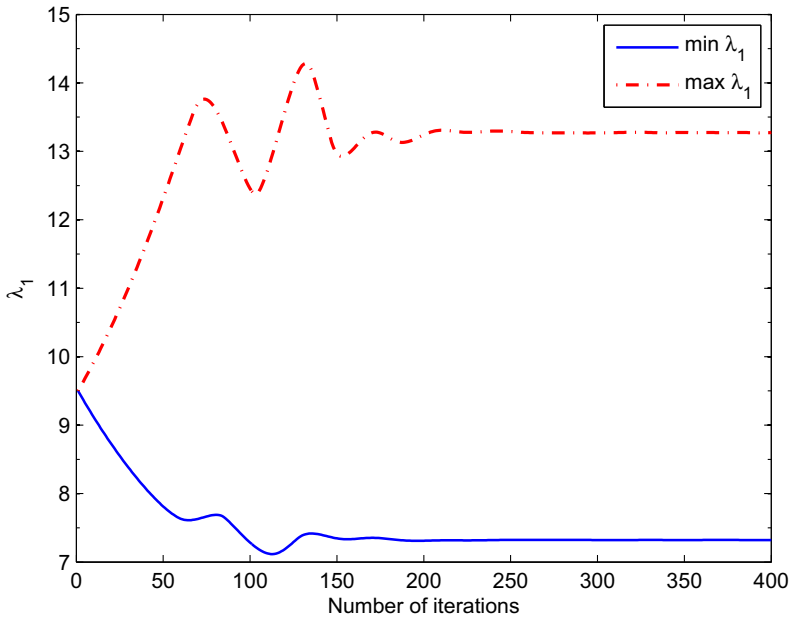


Fig. 3. Convergence history of λ_1 by Algorithm 1.

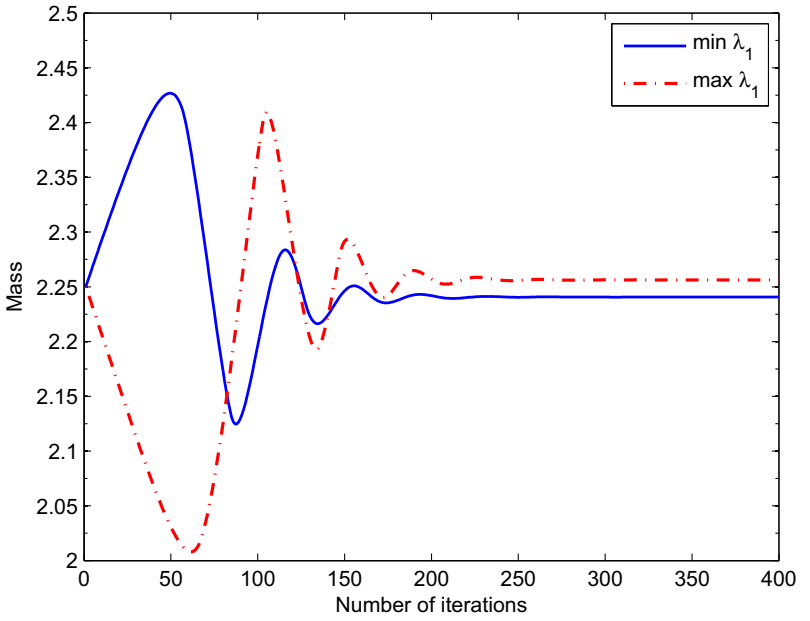


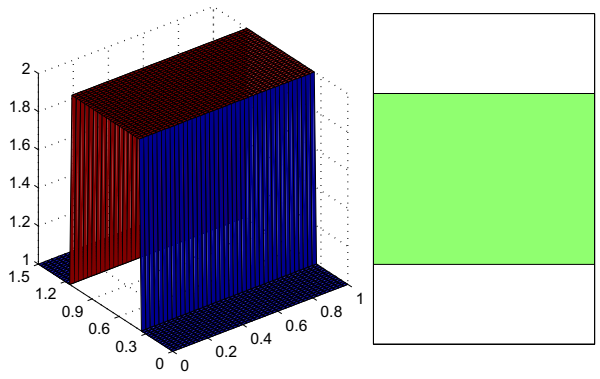
Fig. 4. Mass history for minimization and maximization of λ_1 by Algorithm 1.

$$\bar{\phi} \equiv \operatorname{sgn}(\phi - 1.5) = \begin{cases} \frac{\phi-1.5}{|\phi-1.5|} & \text{if } \phi \neq 1.5, \\ 0 & \text{else} \end{cases} \tag{56}$$

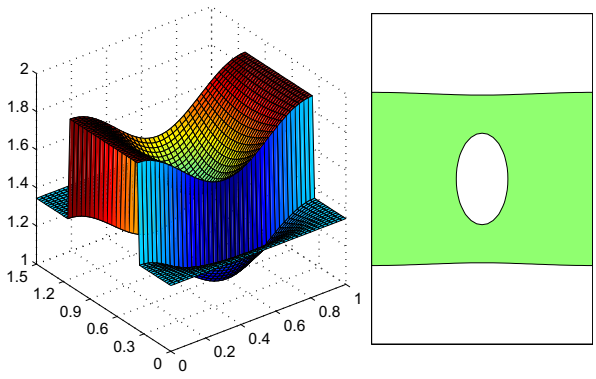
and replace ϕ by the function $\hat{\phi} = \frac{1}{2}\bar{\phi} + 1.5$. Inserting $\hat{\phi}$ into Eq. (12), we obtain

$$\rho(\phi) = \begin{cases} \rho_1\left(\frac{\phi-1.5}{2|\phi-1.5|} + 0.5\right) + \rho_2\left(0.5 - \frac{\phi-1.5}{2|\phi-1.5|}\right) & \text{if } \phi \neq 1.5, \\ \frac{1}{2}(\rho_1 + \rho_2) & \text{else.} \end{cases} \tag{57}$$

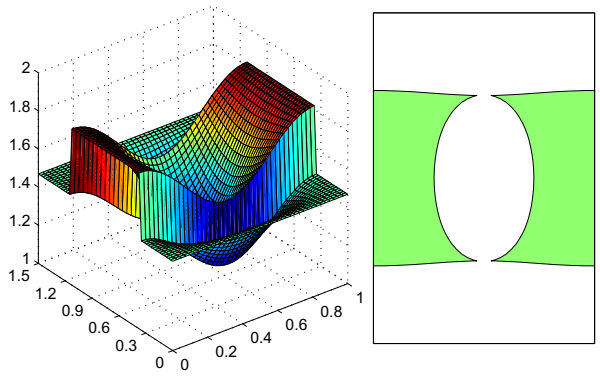
By the chain rule, we have



(a) Initial design



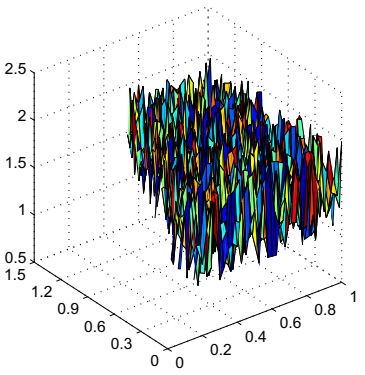
(b) Step 40



(c) Step 60

$$\frac{\partial \rho}{\partial \phi} = \frac{\partial \rho}{\partial \hat{\phi}} \frac{\partial \hat{\phi}}{\partial \bar{\phi}} \frac{\partial \bar{\phi}}{\partial \phi} = \frac{1}{2} \delta(\phi - 1.5) \frac{\partial \rho}{\partial \hat{\phi}}, \tag{58}$$

where δ denotes the Dirac delta function, i.e., $\delta(0) = 1$ and $\delta(x) = 0\ \forall x \neq 0$.



In numerical implementations, $\hat{\phi}$ is replaced by a smoothed approximation:

$$\hat{\phi} \approx \frac{\phi - 1.5}{2\sqrt{(\phi - 1.5)^2 + \epsilon}} + 1.5, \tag{59}$$

where ϵ is a small positive number which has to be chosen properly. As ϕ is replaced by $\hat{\phi}$, the gradient computation in (47) should also be changed using (58). In numerical experiments, however, we observe that satisfying results are obtained if we

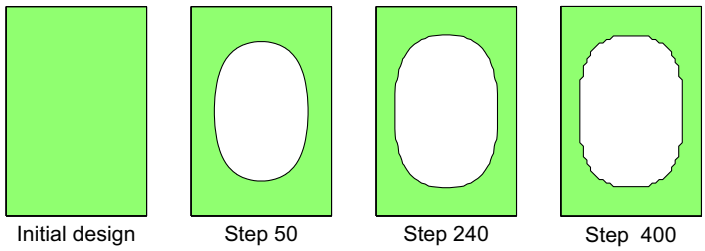


Fig. 7. Minimization of λ_1: evolution of the densities from initial design 3.

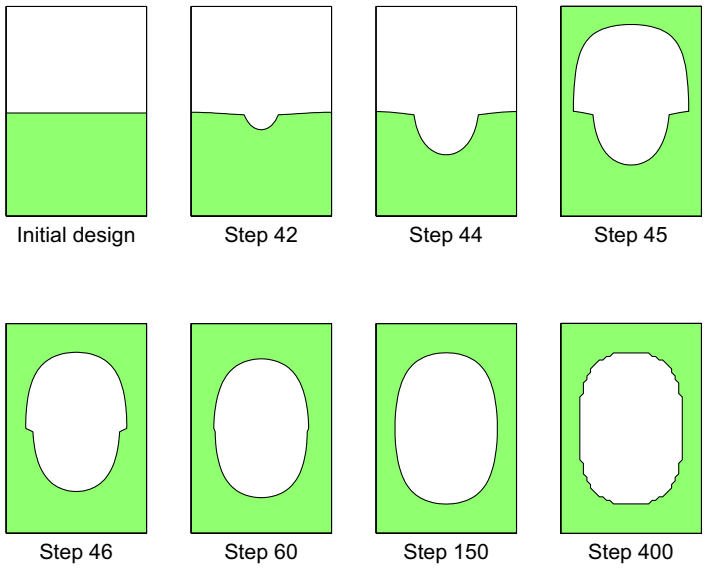


Fig. 8. Minimization of λ_1: evolution of the densities from initial design 4.

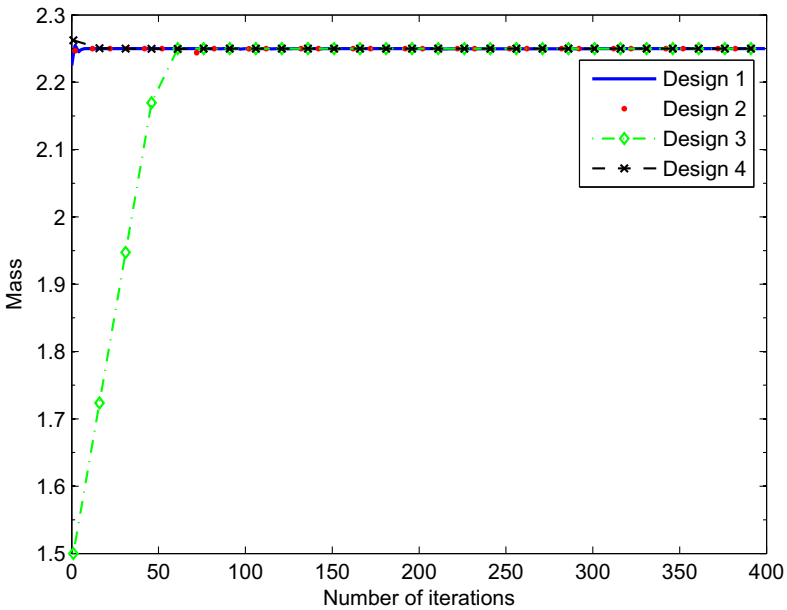


Fig. 9. Minimization of λ_1 with four initial designs: mass conservation.

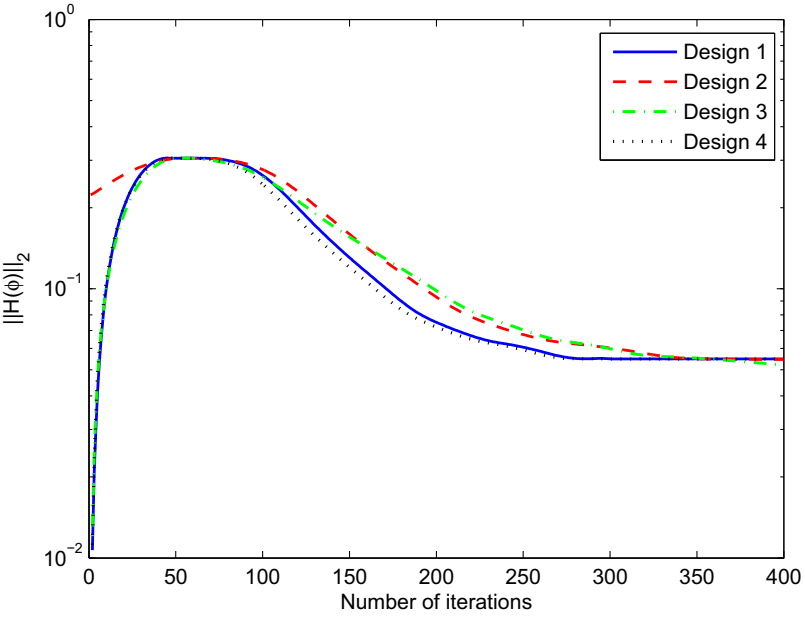


Fig. 10. Minimization of λ_1 with four initial designs: $\|H(\phi)\|_2$.

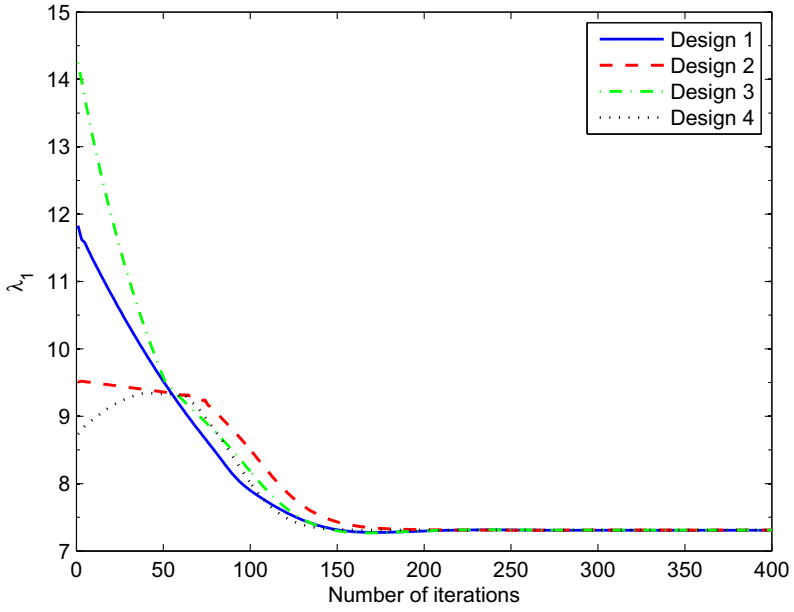
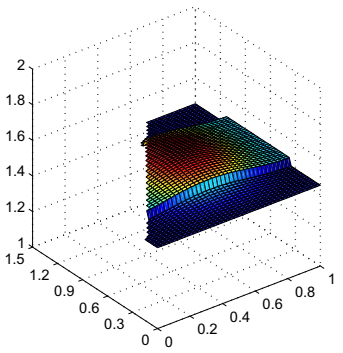


Fig. 11. Minimization of λ_1 with four initial designs: convergence history of λ_1.

Table 1
Effect of initial designs on $\min \lambda_1$ and $\|H(\phi)\|_2$ by Algorithm 2.

Initial design	λ_1	$\|H(\phi)\|_2$
Fig. 5(a)	7.3073	5.49×10^{-2}
Fig. 6(a)	7.3091	5.46×10^{-2}
Fig. 7(a)	7.3085	5.22×10^{-2}
Fig. 8(a)	7.3094	5.49×10^{-2}



just replace $\delta(\phi - 1.5)/2$ by 1. We have chosen 0.01 as the initial value for ϵ and decreased it by a factor 0.98 in each iteration.

In the following text, we refer to the ordinary and the accelerated implementation as the ordinary projection Lagrangian method (OPLM) and the accelerated projection Lagrangian method (APLM), respectively.

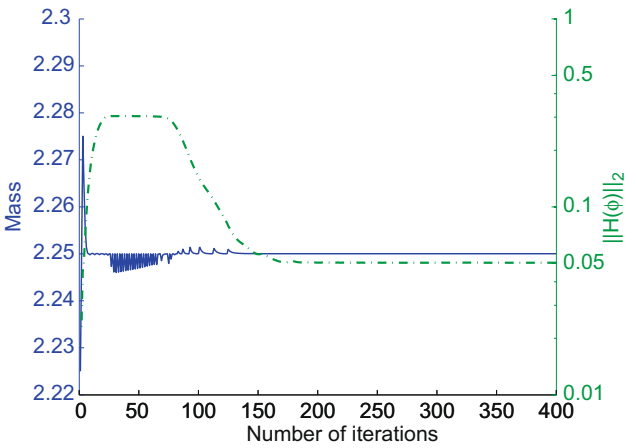
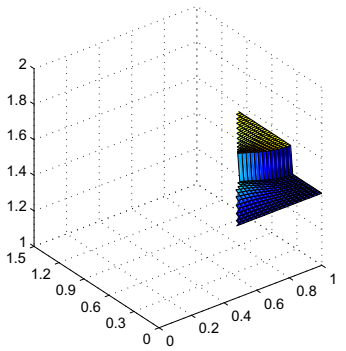


Fig. 14. Maximization of λ_1: mass and $\|H(\phi)\|_2$.



4.3.1. Case 1: $\min \lambda_1$

We test the OPLM for $\min \lambda_1$. Evolutions of the LSF and the densities from four different initial designs are shown in Figs. 18–21, which show that Algorithm 3 is nearly independent of the initial design or the initial LSF ϕ. Figs. 22 and 23 show that the two constraints can be satisfied well. In Fig. 24, we can see clearly that Algorithm 3 enjoys fast convergence. In Table 3, we report the values of λ_1 and $\|H(\phi)\|_2$ after 400 iteration steps. All the four designs lead to $\lambda_1 \approx 7.366$, which is slightly larger than that obtained by Algorithm 2. But the values of $\|H(\phi)\|_2$ are a little smaller than those obtained by Algorithm 2 since the LSF in the region away from the interface reaches to 1 and 2 more exactly. However, the LSF defined near the interface forms the discontinuity more slowly than in Algorithm 2 due to no penalization effect on the constraint.

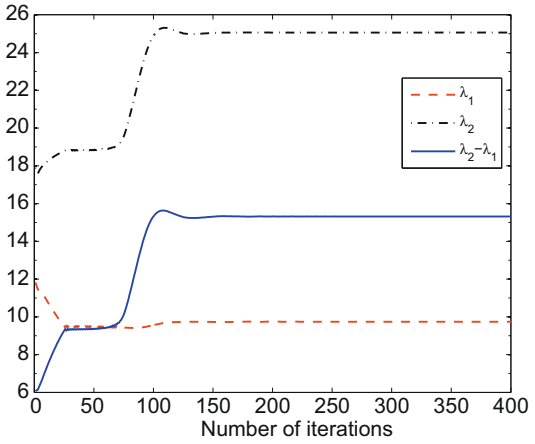


Fig. 16. Maximization of $\lambda_2 - \lambda_1$: convergence history. After 400 iterations, $\lambda_2 - \lambda_1 = 15.3173$.

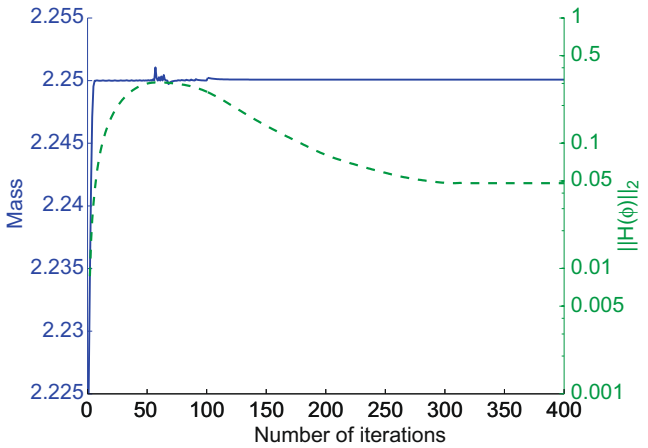


Fig. 17. Maximization of $\lambda_2 - \lambda_1$: mass and $\|H(\phi)\|_2$.

Table 2
Comparison of Algorithm 2 and LSM [37] on the optimal values of λ_1 and $\lambda_2 - \lambda_1$.

Problem	Algorithm 2	LSM [37]
min λ_1	7.31	7.37
max λ_1	13.32	13.45
max $\lambda_2 - \lambda_1$	15.32	15.46

4.3.2. Case 2: max λ_1

The second example demonstrates the process of maximizing the first eigenvalue by the OPLM. The corresponding evolution of the density distribution is shown in Fig. 25. The initial design is similar as in [37], but the evolution process is different in the topological changes. The algorithm reaches the maximum eigenvalue at a little above 13.2 (see Fig. 26) after 400 iterations. The value of λ_1 keeps constant only after 100 iterations, but $\|H(\phi)\|_2$ still decreases rather slowly after 400 iterations.

From Case 1 and Case 2, we conclude that the computed optimal shape can be determined approximately in a few steps. Correspondingly, the mass quantity and the optimal value of the first eigenvalue also reach the stable state quickly. However, ϕ cannot converge to 1 and 2 rigorously around the interface. We have chosen $\sigma = 2$ in the above two cases. It should be noted that this value for time step choosing is larger than those in Algorithms 1 and 2. In the next case, we use the APLM to speed up the formation of the discontinuity.

4.3.3. Case 3: $\max(\lambda_{m+1} - \lambda_m), m = 1,2,3,4$

Next we test the APLM for spectral gap from $\lambda_2 - \lambda_1$ to $\lambda_5 - \lambda_4$. For optimizing $\lambda_2 - \lambda_1$, we set $\sigma = 1.5$ and use the same initial distribution as in [23,37]. Evolution of densities shown in Fig. 27 is different from that by the LSM [23,37]. We use

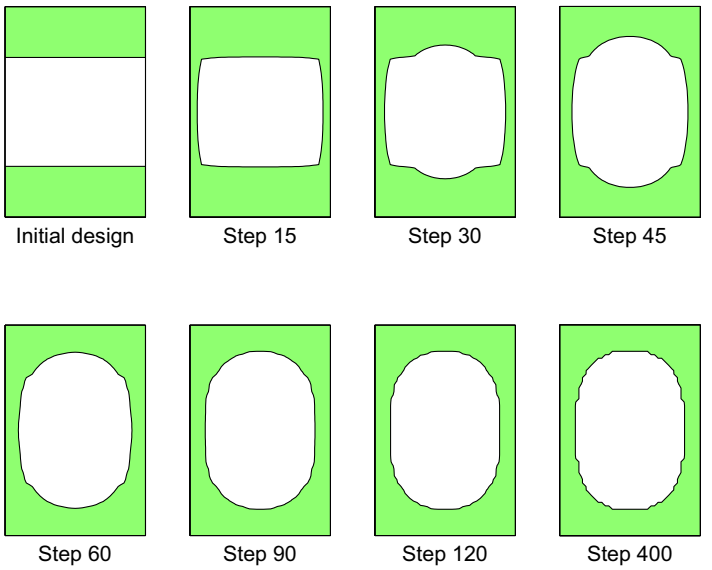


Fig. 18. Minimization of λ_1: evolution of the densities from initial design 1.

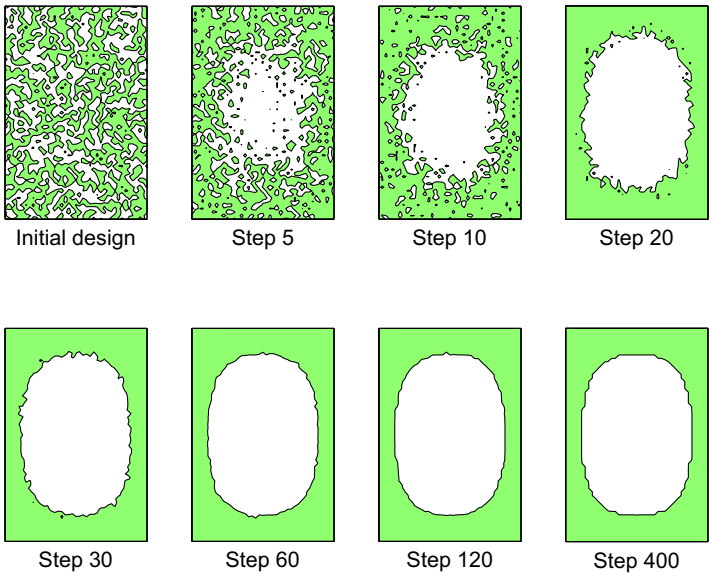


Fig. 19. Minimization of λ_1: evolution of the densities from initial design 2.

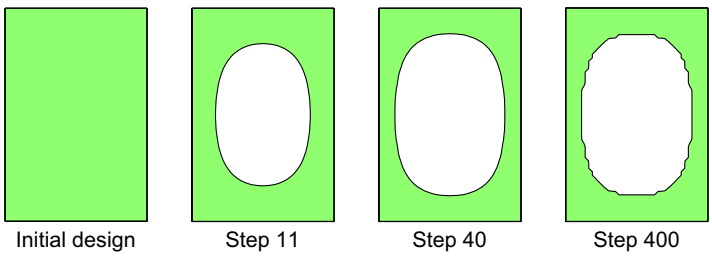


Fig. 20. Minimization of λ_1: evolution of the densities from initial design 3.

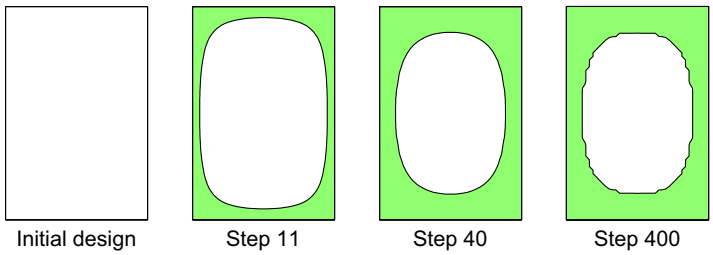


Fig. 21. Minimization of λ_1: evolution of the densities from initial design 4.

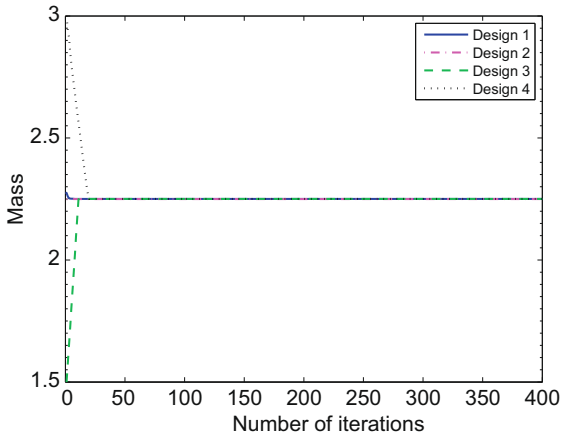


Fig. 22. Mass curves for minimization of λ_1 with four initial designs.

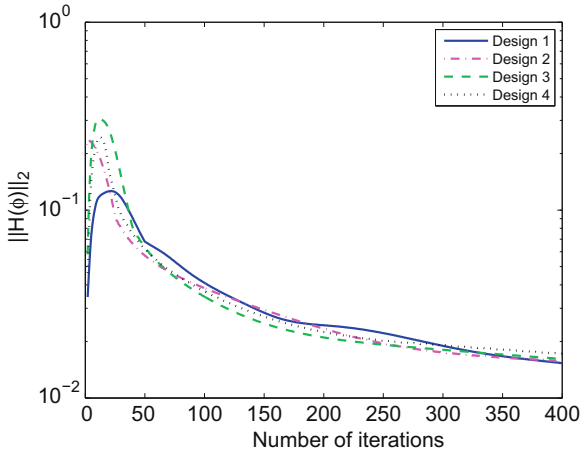


Fig. 23. $||H(\phi)||_2$ for minimization of λ_1 with four initial designs.

the OPLM in the first 300 iterations. Then we apply the APLM to speed up convergence. The evolution of the gap as we iterate is shown in Fig. 28. Fig. 29 illustrates that the piecewise constant constraint can be satisfied much more accurately by using APLM compared with OPLM. The converged value for $\|H(\phi)\|_2$ can be improved from approximately 10^{-2} to 10^{-7}.

For maximizing $\lambda_3 - \lambda_2$, we choose a fixed $\Delta t = 0.0015$ during the iterations. See Figs. 30 and 31 for evolutions of densities and the spectral gap, respectively. Fig. 31 shows clearly that using the APLM can improve the gap value when the OPLM reaches static state.

For optimizing $\lambda_4 - \lambda_3$, the LSM without topological derivatives gets stuck in a local maximum as shown in [23]. As shown in Fig. 32 for evolution of densities with $\Delta t = 0.001$, both of OPLM and APLM can easily create new holes as the LSM based on

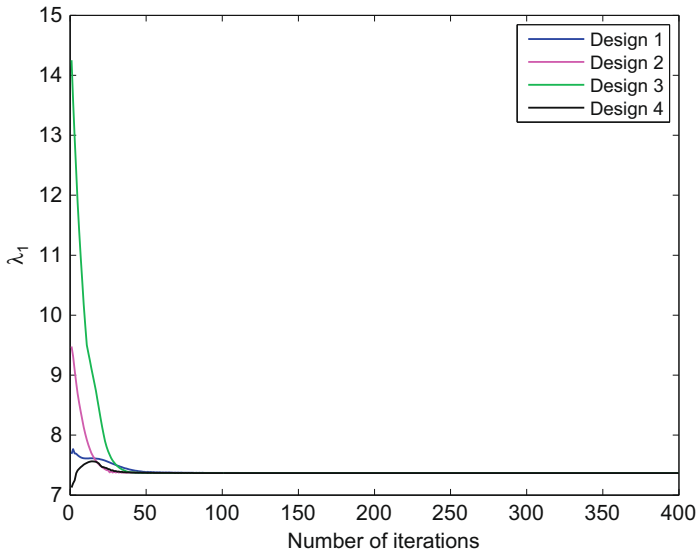


Fig. 24. Convergence history for minimization of λ_1 with four initial designs. See Figs. 18–21 for corresponding density evolutions.

Table 3
Effect of initial designs on $\min \lambda_1$ and $\|H(\phi)\|_2$ by OPLM.

Initial design	λ_1	$\|H(\phi)\|_2$
Fig. 18(a)	7.3669	1.53×10^{-2}
Fig. 19(a)	7.3664	1.58×10^{-2}
Fig. 20(a)	7.3666	1.61×10^{-2}
Fig. 21(a)	7.3665	1.72×10^{-2}

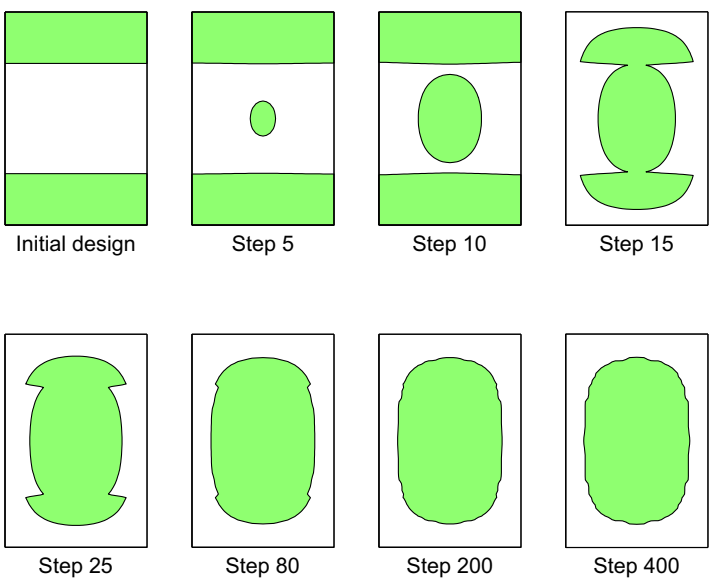


Fig. 25. Maximization of λ_1: evolution of the densities.

combination of shape derivatives and topological derivatives in [23]. From Fig. 33, we see the necessity of using APLM to speed up convergence and improve the gap value.

For maximizing $\lambda_5 - \lambda_4$, we set $\sigma = 2$ and use a different initial guess, which is shown in the upper left corner of Fig. 34. From Fig. 34, we see the topology of the optimal design appears quickly. In Fig. 35, the final density distribution and gap value agree well with those obtained by LSMs [23,37].

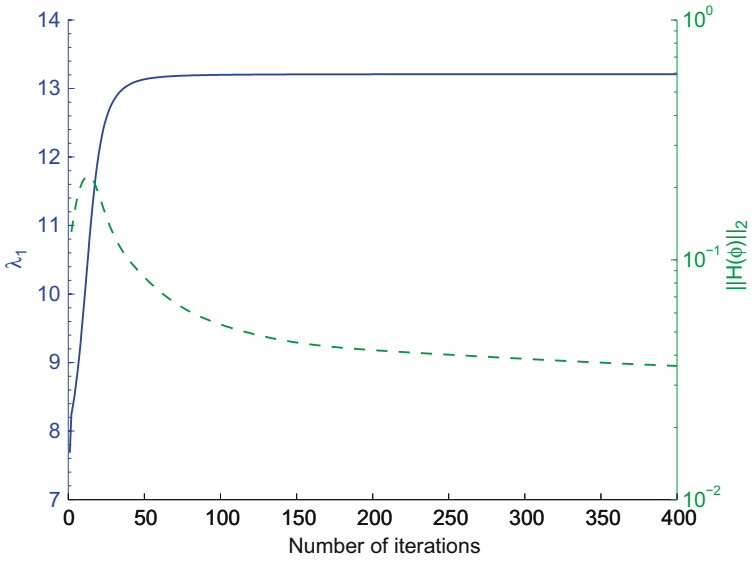


Fig. 26. Convergence history for $\max \lambda_1$ and $\|H(\phi)\|_2$ by OPLM. After 400 iterations, $\lambda_1 = 13.2093$ and $\|H(\phi)\|_2 = 0.0361$.

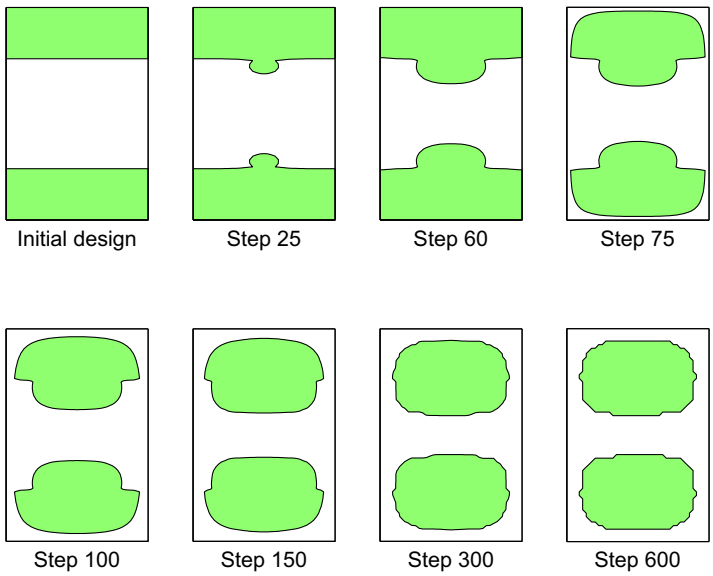


Fig. 27. Maximization of $\lambda_2 - \lambda_1$ by APLM: evolution of the densities.

In Table 4, we compare APLM with LSMs on the optimal band gap values. Most of our results are slightly lower than those obtained using level set approaches. This is mainly because the accuracy of the proposed algorithms based on PCLSM is not sub-pixel. However, compared with the conventional LSM, our algorithms are generally less possible to fall into local optimum.

Remark 7. Both of OPLM and APLM nearly have no requirement for the initial LSF, while the traditional LSM needs to initialize the level set function to a signed distance function.

Remark 8. The algorithms proposed here based on the piecewise constant level set framework can merge, break, and form complicated geometries with sharp corners during the evolution of the LSF as the traditional LSM. Moreover, it can create new holes automatically like the LSM with topological derivatives, while the conventional LSM cannot.

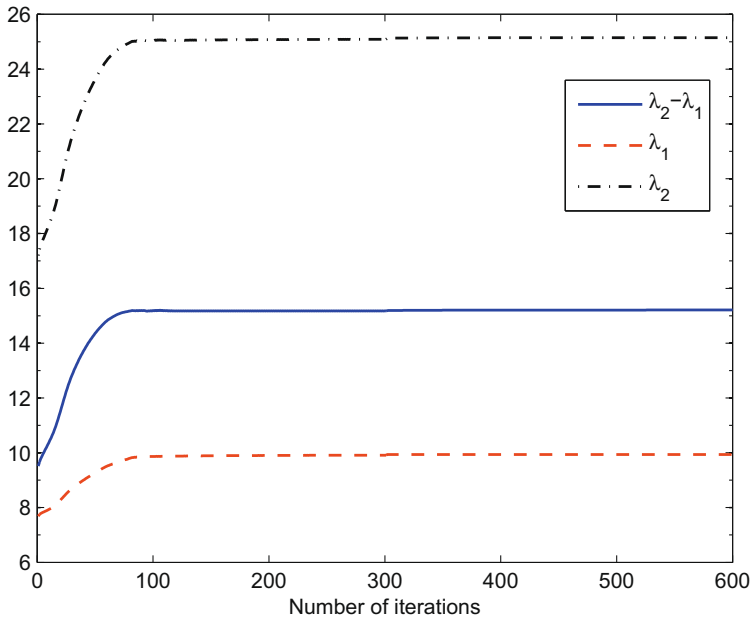


Fig. 28. Convergence history for maximization of $\lambda_2 - \lambda_1$ by APLM. After 300 iterations, $\lambda_2 - \lambda_1 = 15.1866$. After 600 iterations, $\lambda_2 - \lambda_1 = 15.2258$.

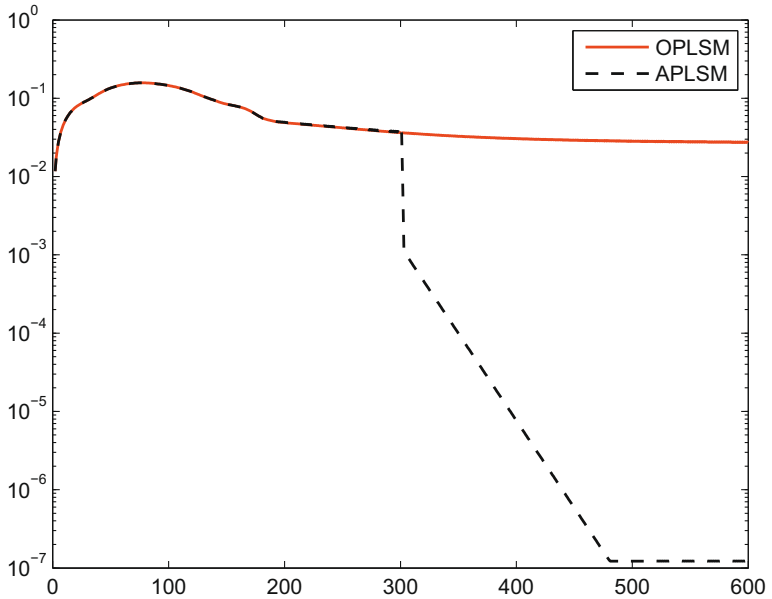


Fig. 29. $\|H(\phi)\|_2$ for maximization of $\lambda_2 - \lambda_1$ by OPLM and APLM.

Remark 9. Our algorithms are effective and efficient for the model problem, but they are by no means free of flaws. There exists a minor defect on the sub-cell resolution of the phase-interface. This is a common problem in the PCLSM while the continuous LSM does not have. Our algorithms lead to a zigzag interface of the final two-phase design compared to a relatively smooth interface obtained by the LSM.

5. Conclusions and future work

In this paper, we have applied the PCLSM to solve the two-density drum design problem. We first presented the augmented Lagrangian method for the model problem and used the Uzawa algorithm to solve it. In order to overcome the lim-

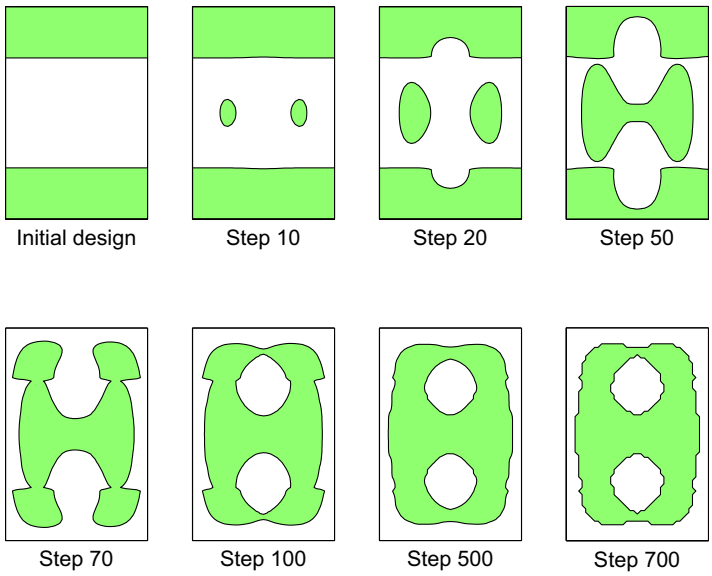


Fig. 30. Maximization of $\lambda_3 - \lambda_2$ by APLM: evolution of the densities.

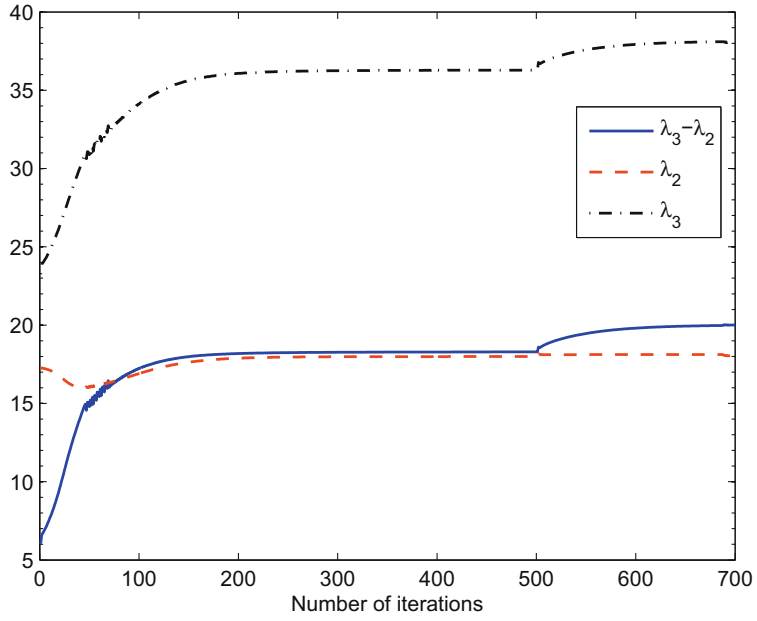


Fig. 31. Convergence history for maximization of $\lambda_3 - \lambda_2$ by APLM. After 500 iterations, $\lambda_3 - \lambda_2 = 18.2892$. After 700 iterations, $\lambda_3 - \lambda_2 = 20.0648$.

itations of the algorithm, the Lagrange multiplier approach for the area constraint was combined with the augmented Lagrangian method for the piecewise constant constraint to form a novel hybrid algorithm. This algorithm is more accurate in geometry constraint satisfaction and more robust to initial guess than the first algorithm. The last new algorithm was given by coupling the Lagrange multiplier method and the projection Lagrangian method. The restriction on the time step in the former two algorithms is relaxed. An acceleration technique was proposed to speed up the formation of discontinuity. All of the three algorithms presented are demonstrated to be effective in numerical simulation for the model problem.

Numerical results show that the methodology proposed here is an interesting alternative to level set methods for solving optimal shape design and topology optimization problems. The conventional LSM based on shape sensitivity analysis cannot create new holes automatically during the evolution which may get stuck at shapes with fewer holes than the optimal geom-

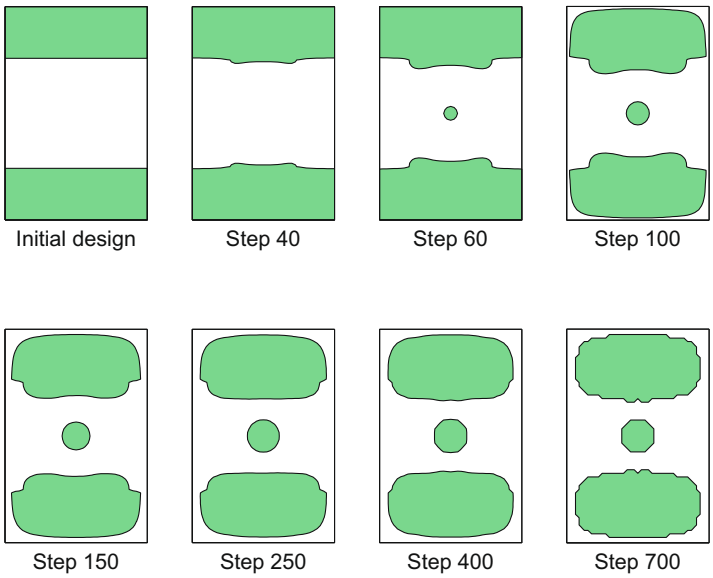


Fig. 32. Maximization of $\lambda_4 - \lambda_3$ by APLM: evolution of the densities.

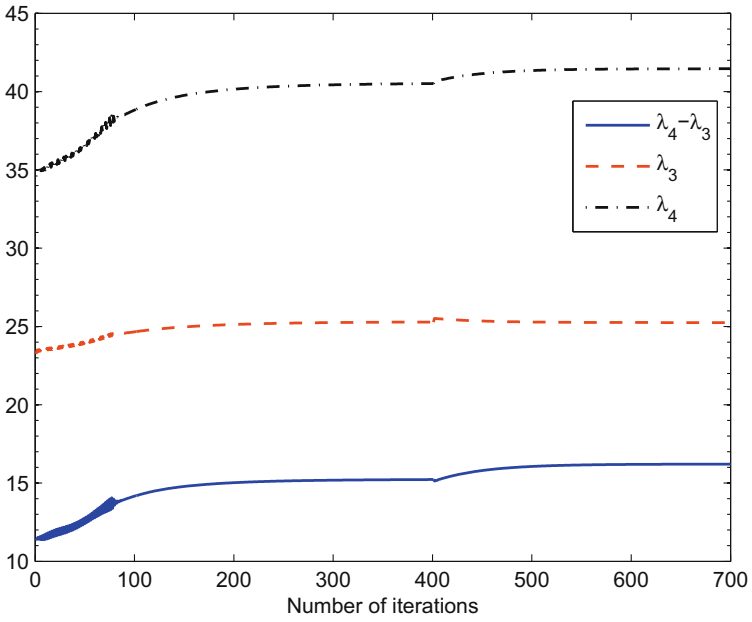


Fig. 33. Convergence history for maximization of $\lambda_4 - \lambda_3$ by APLM. After 400 iterations, $\lambda_4 - \lambda_3 = 15.2549$. After 700 iterations, $\lambda_4 - \lambda_3 = 16.2447$.

etry. By the variational piecewise constant level set approach, topological changes especially the nucleation of small holes are allowed during the iterative process. The initial guess for the geometry of the optimal shape is not needed in our methods. Moreover, we do not need to solve the Hamilton–Jacobi equation and eliminate the re-initialization procedure in the conventional LSM. However, compared with LSMs, the accuracy of the proposed algorithms is not sub-pixel, which is a common limitation of the PCLSM.

The presented algorithms can be improved further although they perform well enough for the model problem. For the gradient descent flow, we will use implicit and semi-implicit schemes to increase the time step size. Besides, Newton-type methods, such as the Quasi-Newton method can also be employed to accelerate convergence. We expect our robust and promising algorithms to be applied to other two-phase or multiphase shape optimization problems and inverse problems in 2D and 3D in the near future.

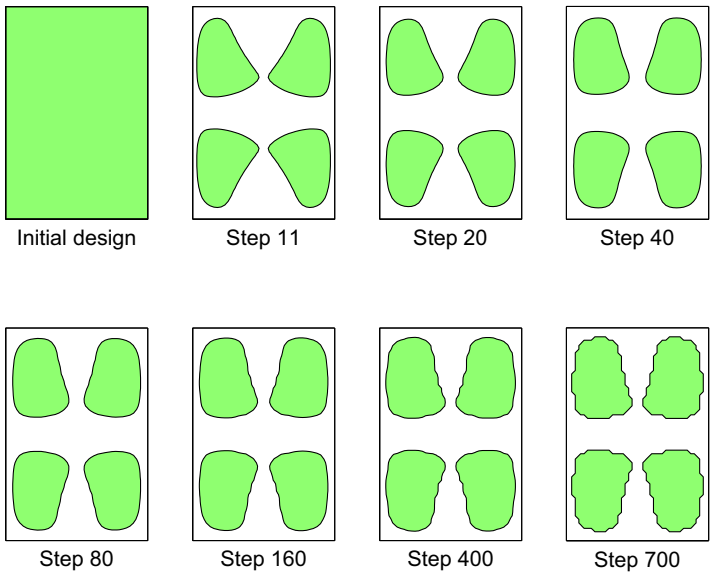


Fig. 34. Maximization of $\lambda_5 - \lambda_4$ by APLM: evolution of the densities.

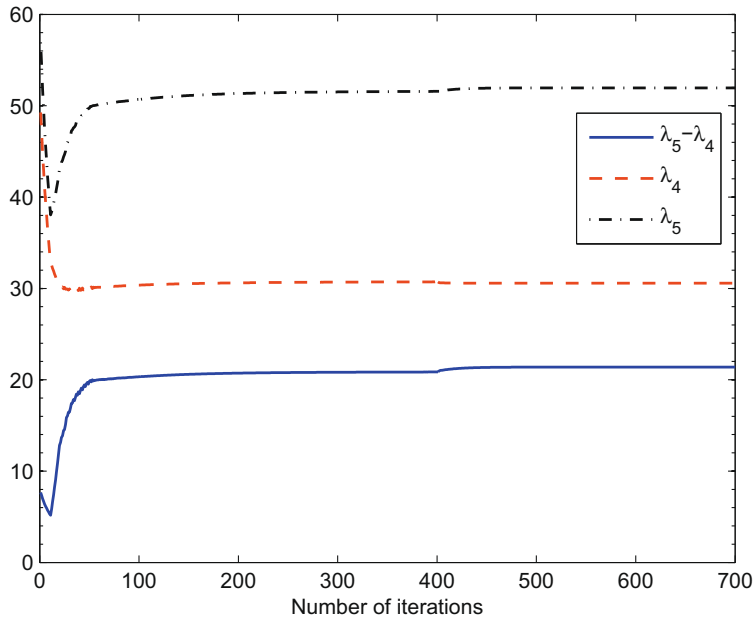


Fig. 35. Convergence history for maximization of $\lambda_5 - \lambda_4$ by APLM. After 400 iterations, $\lambda_5 - \lambda_4 = 20.8667$. After 700 iterations, $\lambda_5 - \lambda_4 = 21.4153$.

Table 4
Comparison of APLM and LSMs on optimal band gaps.

Band gap	APLM	LSM[a]	LSMT[b]
$\lambda_2 - \lambda_1$	15.2258	15.4618	15.4724
$\lambda_3 - \lambda_2$	20.0648	20.1239	20.1600
$\lambda_4 - \lambda_3$	16.2447	15.2323	16.3642
$\lambda_5 - \lambda_4$	21.4153	21.5945	21.4818

[a] LSM without topological derivatives [37].
[b] LSM with topological derivatives [23].

Acknowledgments

The authors express their sincere thanks to Prof. Osher and anonymous reviewers for their valuable comments and constructive suggestions, which have greatly improved the original version of the paper. This work was supported by National Natural Science Foundation of China (No. 10871178) and Natural Science Foundation (No. Y606154) of Zhejiang Province.

References

[1] G. Allaire, Shape Optimization by the Homogenization Method, Springer Verlag, New York, 2001.
[2] G. Allaire, F. Jouve, A. Toader, Structural optimization using sensitivity analysis and a level-set method, J. Comput. Phys. 194 (2004) 363–393.
[3] G. Allaire, F. Jouve, A. Toader, A level set method for shape optimization, C.R. Acad. Sci. Paris, Ser. I 334 (2002) 1125–1130.
[4] G. Allaire, F. De Gournay, F. Jouve, A.M. Toader, Structural optimization using topological and shape sensitivity via a level set method, Control Cybernet. 34 (2005) 59–80.
[5] M. Bendsøe, O. Sigmund, Topology Optimization: Theory, Methods and Applications, Springer Verlag, Berlin, Heidelberg, 2003.
[6] D.P. Bertsekas, Constrained Optimization and Lagrange Multiplier Methods, Academic Press, New York, 1982.
[7] J. Brandman, A level-set method for computing the eigenvalues of elliptic operators defined on compact hypersurfaces, J. Sci. Comput. 37 (2008) 282–315.
[8] M. Burger, A level set method for inverse problems, Inverse Probl. 17 (2001) 1327–1355.
[9] M. Burger, A framework for the construction of level set methods for shape optimization and reconstruction, Interf. Free Bound. 5 (2003) 301–329.
[10] M. Burger, B. Hackl, W. Ring, Incorporating topological derivatives into level set methods, J. Comput. Phys. 194 (2004) 344–362.
[11] M. Burger, S. Osher, A survey on level set methods for inverse problems and optimal design, Eur. J. Appl. Math. 16 (2005) 263–301.
[12] M. Burger, S. Osher, E. Yablonovitch, Inverse problem techniques for the design of photonic crystals, IEICE Trans. Electron. 87 (2004) 258–265.
[13] T.F. Chan, X.-C. Tai, Level set and total variation regularization for elliptic inverse problems with discontinuous coefficients, J. Comput. Phys. 193 (2003) 40–66.
[14] T.F. Chan, X.-C. Tai, Identification of discontinuous coefficients in elliptic problems using total variation regularization, SIAM J. Sci. Comput. 25 (2003) 881–904.
[15] F. Chantalat, C.-H. Bruneau, C. Galusinski, A. Iollo, Level-set, penalization and Cartesian meshes: a paradigm for inverse problems and optimal design, J. Comput. Phys. 228 (2009) 6291–6315.
[16] S.J. Cox, D.C. Dobson, Maximizing band gaps in two-dimensional photonic crystals, SIAM J. Appl. Math. 59 (1999) 2108–2120.
[17] S.J. Cox, D.C. Dobson, Band structure optimization of two-dimensional photonic crystals in *H*-polarization, J. Comput. Phys. 158 (2000) 214–224.
[18] M.-R. Cui, Analysis of iterative algorithms of Uzawa type for saddle point problems, Appl. Numer. Math. 50 (2) (2004) 133–146.
[19] R. Fedkiw, G. Sapiro, C.-W. Shu, Shock capturing, level sets, and PDE based methods in computer vision and image processing: a review of Osher's contributions, J. Comput. Phys. 185 (2003) 309–341.
[20] P. Fulmanski, A. Laurain, J.-F. Scheid, J. Sokołowski, Level set method with topological derivatives in shape optimization, Int. J. Comp. Math. 85 (2008) 1491–1514.
[21] F. Gibou, R. Fedkiw, A fast hybrid *k*-means level set algorithm for segmentation, in: Proc. 4th Annu. Hawaii Int. Conf. Statistics and Mathematics, 2006, pp. 281–291.
[22] E. Haber, A multilevel, level-set method for optimizing eigenvalues in shape design problems, J. Comput. Phys. 198 (2004) 518–534.
[23] L. He, C.-Y. Kao, S. Osher, Incorporating topological derivatives into shape derivatives based level set methods, J. Comput. Phys. 225 (2007) 891–909.
[24] M. Hintermüller, W. Ring, A second order shape optimization approach for image segmentation, SIAM J. Appl. Math. 64 (2003) 442–467.
[25] G. Jiang, D. Peng, Weighted ENO schemes for Hamilton–Jacobi equations, SIAM J. Sci. Comput. 21 (1999) 2126–2143.
[26] K. Kunisch, X.-C. Tai, Sequential and parallel splitting methods for bilinear control problems in Hilbert spaces, SIAM J. Numer. Anal. 34 (1997) 91–118.
[27] A. Leitão, O. Scherzer, On the relation between constraint regularization, level sets, and shape optimization, Inverse Probl. 19 (2003) 1–11.
[28] H. Li, X.-C. Tai, Piecewise constant level set method to interface problems, UCLA CAM Report 06-05, 2006.
[29] J. Lie, M. Lysaker, X.-C. Tai, A piecewise constant level set framework, Int. J. Numer. Anal. Model. 2 (2005) 422–438.
[30] J. Lie, M. Lysaker, X.-C. Tai, A binary level set model and some applications to Mumford–Shah image segmentation, IEEE Trans. Image Process. 15 (2006) 1171–1181.
[31] J. Lie, M. Lysaker, X.-C. Tai, A variant of the level set method and applications to image segmentation, Math. Comput. 75 (2006) 1155–1174.
[32] Z. Luo, L. Tong, J. Luo, P. Wei, M.Y. Wang, Design of piezoelectric actuators using a multiphase level set method of piecewise constants, J. Comput. Phys. 228 (2009) 2643–2659.
[33] L.K. Nielsen, X.-C. Tai, S.I. Aanonsen, M. Espedal, A binary level set model for elliptic inverse problems with discontinuous coefficients, Int. J. Numer. Anal. Model. 4 (2007) 74–99.
[34] J. Nocedal, S. Wright, Numerical Optimization, Springer Verlag, New York, 1999.
[35] S. Osher, R. Fedkiw, Level set methods: an overview and some recent results, J. Comput. Phys. 169 (2001) 463–502.
[36] S. Osher, R. Fedkiw, Level Set Methods and Dynamic Implicit Surfaces, Springer, 2003.
[37] S. Osher, F. Santosa, Level set methods for optimization problems involving geometry and constraints I. Frequencies of a two-density inhomogeneous drum, J. Comput. Phys. 171 (2001) 272–288.
[38] S. Osher, J.A. Sethian, Fronts propagating with curvature dependent speed: algorithms based on Hamilton–Jacobi formulations, J. Comput. Phys. 79 (1988) 12–49.
[39] S. Osher, C.-W. Shu, High-order essentially non-oscillatory schemes for Hamilton–Jacobi equations, SIAM J. Numer. Anal. 28 (1991) 907–922.
[40] P.-O. Persson, G. Strang, A simple mesh generator in matlab, SIAM Rev. 46 (2004) 329–345.
[41] G. Strang, P.-O. Persson, Circuit simulation and moving mesh generation, IEIC Technical Report (Institute of Electronics, Information and Communication Engineers) 104 (2004) 19–24.
[42] G.I.N. Rozvany (Ed.), Topology Optimization in Structural Mechanics, Springer Verlag, New York, 1997.
[43] L. Rudin, S. Osher, E. Fatemi, Nonlinear total variation based noise removal algorithms, Physica D 60 (1992) 259–268.
[44] J.A. Sethian, Level Set Methods and Fast Marching Methods, second ed., Cambridge University Press, Cambridge, 1999.
[45] J.A. Sethian, P. Smereka, Level set methods for fluid interfaces, Annu. Rev. Fluid Mech. 35 (2003) 341–372.
[46] J.A. Sethian, A. Wiegmann, Structural boundary design via level set method and immersed interface methods, J. Comput. Phys. 163 (2000) 489–528.
[47] J. Sokołowski, A. Żochowski, On the topological derivative in shape optimization, SIAM J. Control Optim. 37 (1999) 1251–1272.
[48] J. Sokołowski, J.-P. Zolesio, Introduction to Shape Optimization: Shape Sensitivity Analysis, Springer, Heidelberg, 1992.
[49] B. Song, T.F. Chan, A fast algorithm for level set based optimization, UCLA CAM Report 02-68, 2002.
[50] M. Sussman, P. Smereka, S. Osher, A level set approach for computing solutions to incompressible two-phase flow, J. Comput. Phys. 114 (1994) 146–159.
[51] X.-C. Tai, T.F. Chan, A survey on multiple level set methods with applications for identifying piecewise constant functions, Int. J. Numer. Anal. Model. 1 (2004) 25–48.
[52] X.-C. Tai, O. Christiansen, Image segmentation using some piecewise constant level set methods with MBO type of projection, Int. J. Comp. Vis. 73 (2007) 61–76.

[53] X.-C. Tai, H. Li, A piecewise constant level set method for elliptic inverse problems, Appl. Numer. Math. 57 (2007) 686–696.
[54] X.-C. Tai, C. Wu, Augmented Lagrangian method, dual methods and split Bregman iteration for ROF model, UCLA CAM Report 09-05, 2009.
[55] Y.-H.R. Tsai, L.-T. Cheng, S. Osher, H.-K. Zhao, Fast sweeping algorithms for a class of Hamilton–Jacobi equations, J. Comput. Phys. 127 (1996) 179–195.
[56] S.Y. Wang, K.M. Lim, B.C. Khoo, M.Y. Wang, An extended level set method for shape and topology optimization, J. Comput. Phys. 221 (2007) 395–421.
[57] M.Y. Wang, X. Wang, D. Guo, A level set method for structural topology optimization, Comput. Meth. Appl. Mech. Eng. 192 (2003) 227–246.
[58] P. Wei, M.Y. Wang, Piecewise constant level set method for structural topology optimization, Int. J. Numer. Meth. Eng. 78 (2009) 379–402.
[59] H.-K. Zhao, T.F. Chan, B. Merriman, S. Osher, A variational level set approach to multiphase motion, J. Comput. Phys. 127 (1996) 179–195.
[60] S. Zhou, Q. Li, A variational level set method for the topology optimization of steady-state Navier–Stokes flow, J. Comput. Phys. 227 (2008) 10178–10195.