Research Article

Two Algorithms for a Class of Elliptic Problems in Shape Optimization

Li Tian, Dai Xiaoxia, and Zhang Chengwei

School of Computing Science, Zhejiang University City College, Hangzhou 310015, China

Correspondence should be addressed to Dai Xiaoxia; xxiadai@163.com

Received 4 May 2015; Accepted 21 June 2015

Academic Editor: Francesco Pellicano

Copyright © 2015 Li Tian et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

We propose two algorithms for elliptic boundary value problems in shape optimization. With the finite element method, the optimization problem is replaced by a discrete variational problem. We give rules and use them to decide which elements are to be reserved. Those rules are determined by the optimization; as a result, we get the optimal design in shape. Numerical examples are provided to show the effectiveness of our algorithms.

1. Introduction

With numerous applications in structural engineering and fluid mechanics, shape optimization has received much more attention in recent years [1–5]. Typical problems are to find optimal shapes that minimize or maximize some design objectives with certain PDE and geometry constraints. The essential difficulty for such problems is that the optimal shape or topology is unknown *a priori*. Many different numerical methods are used to solve this type of problems. The classical method of shape sensitivity analysis has been studied in [6, 7]. One drawback of this method is its computational effort due to remeshing.

The most recent topology approach is the homogenization method [8]. A relaxed formulation of the original shape optimization problem should be introduced. Then, penalization techniques are needed to remove composite designs in the final shape. The method is very promising in that it is nearly independent on the initial guess for design. Moreover, it has a rigorous mathematical result of the existence of the solution. However, this method can be applied to limited areas and particular objective functions.

Level set methods are popular for solving shape optimization problems (see [5, 9–13]). Level set method first proposed by Osher and Sethian [14] is a versatile tool when dealing with problems requiring tracing interfaces separating

a domain into subregions. For details of the method and its wide applications, we refer to the monograph [15] and the survey [16]. Level set methods can easily handle certain types of shape and topological changes, such as merging, splitting, and developing sharp corners, which can be done by the evolution of the level set function. Therefore, level set methods can naturally be applied to solve shape and topology optimization problems. Its cost is moderate since the shape is captured on a fixed Eulerian mesh. However, by its nature, the obtained level set function decreases the number of connected components of resulting geometrical domains during the optimization process. Besides, the convergence to the optimal geometrical domain is slow due to the necessary reinitialization process.

In this paper, we propose a novel method for a class of linear elliptic problems in shape optimization. The main difference from other methods is that we perform optimization in each element, which means that we deal with this problem in a discrete formula. First, let us introduce the model problem. Let U be a bounded domain of R^2 with a boundary $\partial U = \Gamma_D$. Let V also be a subset of R^2 and satisfy $V \subset\subset U$. For any open set $\omega \subset R^2$, we denote by $\#\bar{\omega}$ the number of connected components of $\bar{\omega}$ and we consider the set of admissible domains:

$$\mathcal{O}_k = \{\Omega = U \setminus \bar{\omega}; \omega \subset V, \#\bar{\omega} < k\}. \quad (1)$$

FIGURE 1: An admissible domain Ω .

For any $\Omega \in \mathcal{O}_k$, $k \geq 1$, the boundary of Ω can be denoted by $\partial\Omega = \Gamma_D \cup \Gamma_N$ with $\Gamma_N = \partial\omega$. The boundary Γ_N receives Neumann boundary conditions, while we have Dirichlet conditions on Γ_D . We illustrate the details in Figure 1.

Now, let us consider the following elliptic boundary value problem:

$$\begin{aligned} -\Delta u + u &= f \quad \text{in } \Omega, \\ u &= 0 \quad \text{on } \Gamma_D, \\ \frac{\partial u}{\partial n} &= 0 \quad \text{in } \Gamma_N, \end{aligned} \quad (2)$$

where $f \in C^\infty(\bar{U})$ and n is the unit outwards normal vector to $\partial\Omega$. Our goal is to optimize the shape of domain Ω which maximizes a given cost functional depending on the domain and the solution u . Here, we consider the energy functional:

$$\begin{aligned} E(\Omega, u) &:= \frac{1}{2} \int_{\Omega} (|\nabla u|^2 + u^2) dx - \int_{\Omega} f u dx \\ &= -\frac{1}{2} \int_{\Omega} (|\nabla u|^2 + u^2) dx = -\frac{1}{2} \int_{\Omega} f u dx, \end{aligned} \quad (3)$$

and let us introduce the regularized shape functional:

$$J(\Omega, u) := E(\Omega, u) + \lambda A(\Omega), \quad (4)$$

with $A(\Omega)$ defined by $A(\Omega) := |\Omega|$, which is the Lebesgue measure of Ω in $\mathbb{R}^2$. Constant λ are positive and allow us to take into account the area. Then, for any fixed integer $k \geq 1$, the shape optimization problem is the following:

$$\max \{J(\Omega) : \Omega \in \mathcal{O}_k\}. \quad (5)$$

Since the exterior boundary of Ω is fixed once for all and equals ∂U , the domains Ω satisfying (5) are actually determined by their moving boundaries Γ_N .

Our algorithm comes from greedy algorithm. A greedy algorithm might also be called a “single-minded” algorithm or an algorithm that gobbles up all of its favorites first. It may not completely solve the problem or if it produces a

solution, it may not be the very best one, but it is one way of approaching the problem and sometimes yields very good results. It is well known that greedy algorithms are usually used in discrete mathematics, for example, the salesman problem, graph coloring problem, and other NP-complete problems [17]. And a few papers have reported that the greedy algorithms can also work well for combinatorial optimization problems [18, 19]. In this paper, we propose numerical algorithms based on the greedy algorithm. In one step of our algorithm, we compute the shape functional in each element to find some elements according to the given rules. Then, we throw those elements away (they may come back at some next step). The sum of remain elements is just the domain Ω that is the approximation of our problem.

We will propose two types of algorithms in the next section and present some numerical results in Section 3.

2. Algorithms

In [20], the authors proved that there exists a domain Ω satisfying problem (5). They obtain this result when they apply the results from [21, 22] to this problem with a few changes. Here, the results are listed and the details can be found in [20].

Theorem 1. *Let $\Omega_i = U \setminus \bar{\omega}_i$ be a sequence of open sets such that the number of connected components of $\bar{\omega}_i$ is uniformly bounded. If $\bar{\omega}_i$ converges in the sense of the Hausdorff metric to $\bar{\omega}$, then the solution u_i of the Neumann problem*

$$\begin{aligned} -\Delta u_i + u_i &= f, \quad \text{in } \Omega_i, \\ u_i &= 0 \quad \text{on } \partial U, \\ \partial_n u_i &= 0, \quad \text{on } \partial\omega_i \end{aligned} \quad (6)$$

converges to the solution u of problem (6) on Ω if and only if $|\Omega_i| \rightarrow |\Omega|$.

Theorem 2. *For any $k \geq 0$, problem (5) admits at least one solution $\Omega \in \mathcal{O}_k$.*

```

(1)  $i = 0$ .
(2) Compute the solution  $u_i$  of (8) with domain  $\Omega_i = U \setminus \omega_i$ ,  $SJ = 0$ .
    For  $k = 1$ : Nelement
        Find no marked elements, and compute their shape functional
         $J_i(K) = -(1/2) \int_K f u_i dx + \lambda |K|$ , and the sum of shape functional
         $SJ = SJ + J_i(K)$ ;
    end
    Find the minimum of  $J_i(K)$  and mark the element.
(3)  $i = i + 1$ , Go to Step 2.

```

ALGORITHM 1

```

(1)  $i = 0$ .
(2) Compute the solution  $u_i$  of (8) with domain  $\Omega_i = U \setminus \omega_i$ ,  $SJ = 0$ .
    For  $k = 1$ : Nelement
        Compute  $J_K = -(1/2) \int_K f u_i dx$ ;
        Find no marked elements and compute their sum of shape functional
         $SJ = SJ + J_K + \lambda |K|$ ;
        if  $J_K < -\lambda |K| \theta$ 
            Yes: mark;
            No: remove the mark.
        end
    end
(3)  $i = i + 1$ , Go to Step 2.

```

ALGORITHM 2

For the sake of convenience, we choose $U = (0, 1) \times (0, 1)$. Let $\mathcal{T}_h$ be a partition of the domain U into rectangular elements of mesh size h . Then, we define the bilinear finite element space V_h by

$$V_h = \{v \in H_0^1(U) : v|_k \in Q_1(k), \forall k \in \mathcal{T}_h\}. \quad (7)$$

We take V_h as finite element space of problem (2); then, we can get the following variational formulation: find $u_h \in V_h$ such that

$$\int_{\Omega} (\nabla u_h \cdot \nabla v_h + u_h v_h) dx = \int_{\Omega} f v_h dx, \quad \forall v_h \in V_h. \quad (8)$$

If we change domain Ω into element k , there is no such type of equation set up since the function does not vanish on boundary of each element. In the element K , we obtain a measurement:

$$J(K) = -\frac{1}{2} \int_K f u|_K dx + \lambda |K|. \quad (9)$$

Now, we use the greedy strategy: we drop the element K , in which $J(K) = \min_{\mathcal{K}} J(\mathcal{K})$. Accordingly, we can rewrite the shape functional as the sum of each element; namely,

$$\begin{aligned} J(\Omega) &:= -\frac{1}{2} \int_{\Omega} f u dx + \lambda A(\Omega) \\ &= \sum_k \left(-\frac{1}{2} \int_K f u_k dx + \lambda |K| \right), \end{aligned} \quad (10)$$

where $|K|$ is the area of the element K . In our quadrilateral mesh case, $|K| = h^2$. In this way, we can propose an iterative algorithm; in each step, we seek and throw away the element whose value of shape functional is the minimum. Since each time we throw away only one element, which is very small compared to the whole domain, this algorithm can increase the shape functional in general. We list the algorithm as in Algorithm 1.

Here, “ i ” and “ SJ ” denote the iteration of the algorithm and the value of shape functional on domain Ω_i , respectively. We denote by “*Nelement*” the number of finite elements of domain U . “ ω_i ” represents the sum of elements that are thrown away in the preceding i iterations. When $i = 0$, we solve (8) with the domain U , because we have no marked elements at first. And this is our initial case. From results of the numerical examples (in Section 3), we can see that Algorithm 1 can obtain an optimal domain. Because we throw

FIGURE 2: Three elements.

FIGURE 3: Shape optimization with the data $f = 10 \sin^2(4\pi x)$, $\lambda = 0.5$ using Algorithm 1. Optimal value: $J(\Omega) = 2.7446853$.

FIGURE 4: Shape optimization with the data $f = 10 \sin^2(4\pi x)$, $\lambda = 0.5$, $\theta = 0.7$ using Algorithm 2. Optimal value: $J(\Omega) = 0.278974$.

away one element in each iteration, the convergence is very slow. And the number of iterations gets growth when the mesh size h is refined. What is more, it is difficult to get the conditions for stopping the iterations. Therefore, we make some changes in Algorithm 1. At each iteration, we throw away several elements to accelerate convergence. But the optimal number of elements that should be thrown away is not known *a priori*. Sometimes we may throw away some elements which should be left. Therefore, we should design a process to get such elements back. That is our algorithm (Algorithm 2).

The parameter θ controls how much elements will be thrown away in each step. The optimum of θ is different for different problems. According to this algorithm, in each step, we discard not only one element but a batch of elements in general. Furthermore, the elements which are thrown away maybe come back in some next step. In Figure 2, there are three elements and k_1 is the element and we keep it to Ω_i ; the other two elements are thrown away. In the next step, after solving the equation u_1, u_2 will be changed and therefore the

value of shape functional in element k_2 is changed. Thus, if this value does not satisfy the rule that we set up should be back. This procedure is necessary for our algorithm.

Both of Algorithms 1 and 2 are simple and easy for coding. In the next section, we will see Algorithm 2 enjoys fast convergence for shape optimization problems.

3. Numerical Examples

In this section, we present some numerical tests for our algorithms. As mentioned above, we choose $U = (0, 1) \times (0, 1)$ and use bilinear finite element space V_h as our space. We start with the entire domain U in all examples. In order to make numerical comparisons, we use numerical examples in [20]. The data of the first example are

$$\begin{aligned} f &= 10 \sin^2(4\pi x), \quad \text{for } (x, y) \in U = (0, 1) \times (0, 1), \\ \lambda &= 0.5. \end{aligned} \quad (11)$$

FIGURE 5: Shape optimization with the data $f = 10x + y$, $\lambda = 1$, $\theta = 0.64$ using Algorithm 2. Optimal value: $J(\Omega) = 0.597596$.

First, we apply Algorithm 1 to this example and let $h = 1/64$. We denote by Ω^1 with green the optimal domain obtained in this example. The value of shape functional J in Ω^1 is $J(\Omega^1) = 0.274468$. The results are shown in Figure 3. We find that Algorithm 1 requires many iterations and converges very slowly. Besides, there are some oscillations in iterative process. These two flaws are unfavorable for a good algorithm. In the following, let us see the effects of Algorithm 2. Note that Algorithm 2 involves choice of a parameter θ . Considering the little dependence on h , we choose some θ for a relatively large h and adjust it for a small h . Numerical results are shown in Figure 4.

In [20], in order to make comparisons with the level set method with topological derivatives, conventional level set method was used to solve the model problem. In that case, the Hamilton-Jacobi equation is solved on a 101×101 grid. At last, the level set method converges to a local maximum $J(\Omega) = 0.2640695$. The value $J(\Omega^2) = 0.278974$ obtained by using Algorithm 2 is better than $J(\Omega) = 0.2640695$. Our method is simple and enjoys fast convergence, although the final functional value 0.282319 by the level set method with topological derivatives is a little better than ours.

The second example tests the choice of λ with $f(x, y) = 10x + y$. Thus, we get two groups of figures, Figures 5 and 6. When λ is large, from Figure 6, we see the oscillation exists. This phenomenon is caused by the increase of parameters, which also appears in [20]. Our visible result is similar as that in [20].

4. Conclusion

Based on the greedy idea, we have presented two interesting and promising algorithms for solving the linear elliptic problems in shape optimization. Our algorithms are simple but effective for the model problem. Algorithm 2 is especially efficient compared with the level set method. We will try to apply our method to other shape and topology problems involving geometry and topology in the future.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

FIGURE 6: Shape optimization with the data $f = 10x + y$, $\lambda = 1.7$, $\theta = 0.55$ using Algorithm 2. Optimal value: $J(\Omega) = 1.189091$.

Acknowledgment

Dai Xiaoxia's research was partially supported by the Natural Science Foundation of Zhejiang Province of China under Grants LQ12A01012 and LQ13A010006.

References

- [1] M. P. Bendsøe and O. Sigmund, *Topology Optimization: Theory, Methods and Applications*, Springer, Berlin, Germany, 2003.
- [2] Y. B. Deng, Z. Y. Liu, J. F. Wu, and Y. H. Wu, "Topology optimization of steady Navier-Stokes flow with body force," *Computer Methods in Applied Mechanics and Engineering*, vol. 255, pp. 306–321, 2013.
- [3] A. Myśliński, "Piecewise constant level set method for topology optimization of unilateral contact problems," *Advances in Engineering Software*, vol. 80, pp. 25–32, 2015.
- [4] Y. Q. Wang, Z. Luo, Z. Kang, and N. Zhang, "A multi-material level set-based topology and shape optimization method," *Computer Methods in Applied Mechanics and Engineering*, vol. 283, pp. 1570–1586, 2015.
- [5] B. L. Zhu, X. M. Zhang, and S. Fatikow, "Structural topology and shape optimization using a level set method with distance-suppression scheme," *Computer Methods in Applied Mechanics and Engineering*, vol. 283, pp. 1214–1239, 2015.
- [6] O. Pironneau, *Optimal Shape Design for Elliptic Systems*, Springer, Berlin, Germany, 1984.
- [7] J. Sokolowski and J.-P. Zolesio, *Introduction to Shape Optimization: Shape Sensitivity Analysis*, Springer, Heidelberg, Germany, 1992.
- [8] G. Allaire, *Shape Optimization by the Homogenization Method*, Springer, New York, NY, USA, 2001.
- [9] G. Allaire, F. Jouve, and A.-M. Toader, "Structural optimization using sensitivity analysis and a level-set method," *Journal of Computational Physics*, vol. 194, no. 1, pp. 363–393, 2004.
- [10] X. X. Dai, P. P. Tang, X. L. Cheng, and M. H. Wu, "A variational binary level set method for structural topology optimization," *Communications in Computational Physics*, vol. 13, no. 5, pp. 1292–1308, 2013.
- [11] E. Lee and J. R. A. Martins, "Structural topology optimization with design-dependent pressure loads," *Computer Methods in Applied Mechanics and Engineering*, vol. 233/236, pp. 40–48, 2012.

- [12] Q. Xia, M. Y. Wang, and T. Shi, "Topology optimization with pressure load through a level set method," *Computer Methods in Applied Mechanics and Engineering*, vol. 283, pp. 177–195, 2015.
- [13] Q. Xia, M. Y. Wang, and T. Shi, "A level set method for shape and topology optimization of both structure and support of continuum structures," *Computer Methods in Applied Mechanics and Engineering*, vol. 272, pp. 340–353, 2014.
- [14] S. Osher and J. A. Sethian, "Fronts propagating with curvature-dependent speed: algorithms based on Hamilton-Jacobi formulations," *Journal of Computational Physics*, vol. 79, no. 1, pp. 12–49, 1988.
- [15] S. Osher and R. Fedkiw, *Level Set Methods and Dynamic Implicit Surfaces*, vol. 153 of *Applied Mathematical Sciences*, Springer, New York, NY, USA, 2003.
- [16] S. Osher and R. P. Fedkiw, "Level set methods: an overview and some recent results," *Journal of Computational Physics*, vol. 169, no. 2, pp. 463–502, 2001.
- [17] R. Janczewski, "Greedy T-colorings of graphs," *Discrete Mathematics*, vol. 309, no. 6, pp. 1685–1690, 2009.
- [18] V. V. Shenmaier, "A greedy algorithm for maximizing a linear objective function," *Discrete Applied Mathematics*, vol. 135, no. 1–3, pp. 267–279, 2004.
- [19] Z.-F. Zhang, K.-W. Liang, and X.-L. Cheng, "Greedy algorithms for eigenvalue optimization problems in shape design of two-density inhomogeneous materials," *International Journal of Computer Mathematics*, vol. 88, no. 1, pp. 183–195, 2011.
- [20] P. Fulmansi, A. Laurain, J.-F. Scheid, and J. Sokolowski, "Level set method with topological derivatives in shape optimization," *International Journal of Computer Mathematics*, vol. 85, no. 10, pp. 1491–1514, 2008.
- [21] D. Bucur and N. Varchon, "Boundary variation for a Neumann problem," *Annali della Scuola Normale Superiore di Pisa-Classe di Scienze*, vol. 4, no. 29, pp. 807–821, 2000.
- [22] D. Bucur and N. Varchon, "A duality approach for the boundary variation of Neumann problems," *SIAM Journal on Mathematical Analysis*, vol. 34, no. 2, pp. 460–477, 2002.