

TOPOLOGY OPTIMIZATION WITH AN IMPLICIT FUNCTION AND PARAMETERIZED CUTTING SURFACE

Peter D. Dunning¹

¹ University of Aberdeen
School of Engineering, University of Aberdeen, Aberdeen, AB24 3UE, Scotland
e-mail: peter.dunning@abdn.ac.uk

Keywords: Topology optimization, implicit functions, mathematical programming, design space reduction.

Abstract. *This paper introduces a new implicit function based method for topology optimization that can: obtain solutions with smooth boundaries, be solved using standard mathematical programming methods and reduce the number of design variables. Using implicit functions for topology optimization is attractive because the solutions have clearly defined, smooth boundaries. Most current methods use the zero level-set of the implicit function to define the boundary. The implicit function is then modified during optimization to move the boundary location and connectivity. The new approach proposed in this paper abandons the zero level-set idea and instead uses a fixed signed-distance implicit function. The definition of the boundary from the fixed implicit function is then modified during optimization. This is achieved by using a cutting surface and defining the boundary as the intersection of the cutting surface and signed-distance function. The cutting surface is parameterized and the parameters become the design variables during optimization. Thus, the optimization problem can be solved using mathematical programming and the number of parameters used to define the cutting surface is less than the number of elements in the analysis mesh. The new method is demonstrated using minimization of compliance, minimization of volume and complaint mechanism problems. The results show that the method can obtain good solutions to well-known problems with smooth, clearly defined boundaries and that this can be achieved using significantly fewer design variables compared with element-based methods.*

1 INTRODUCTION

The goal of structural topology optimization is to simultaneously optimize the size, shape and layout (or topology) of a structure. One benefit of this approach, compared with size or shape optimization, is that more efficient designs can be obtained because the optimizer is not constrained by a pre-determined layout.

Many topology optimization methods have emerged over the past 30 years, which have been successfully applied to many applications [1, 2]. There are two main categories of method: those using element-wise design variables, such as the Solid Isotropic Material with Penalization method [3] and Evolutionary Structural Optimization [4], and those using the zero-level set of an implicit functions to represent the boundary [5].

Topology optimization using an implicit function has several advantages over element-based methods, such as solutions with smooth boundaries and avoiding some numerical artifacts, such as checkerboard patterns [5]. Several methods have been developed to exploit these desirable qualities for topology optimization. Some methods use shape derivatives to move the implicit boundary by solving a PDE, such as the conventional level-set and phase-field methods [6, 7]. These approaches often converge slowly, and can require special techniques for solving constrained optimization problems [8]. An alternative approach is to parameterize the implicit function such that the problem can be solved using mathematical programming. This is attractive because advanced algorithms for efficiently solving constrained optimization problems can be employed. Some existing implicit function parameterization methods include, radial basis functions [9], geometric shapes [10], Fourier series [11] and finite element shape functions [12]. Also, projection methods offer similar features [13], as the solutions can have smooth boundaries and the problem can be solved using mathematical programming.

Another potential benefit of using parameterized implicit functions for topology optimization is that the number of design variables may be reduced, compared with methods using element-wise design variables. This is because the design variables are not tied to the finite element discretization used for structural analysis. Reducing the number of variables reduces the problem size, which may result in faster convergence. Furthermore, as the number of design variables decreases optimizers that use full Hessians to speed up convergence, such as IPOPT, become more viable [14]. These advantages are particularly useful for large 3D problems, where a typical element-based method can use 10^6 (or more) design variables [15].

In this paper a new idea for using implicit functions for topology optimization is explored. The idea combines a signed-distance implicit function with a cutting surface. The cutting surface is parameterized and the parameters then become the design variables. The new method is introduced in Section 2, with numerical implementation issues discussed in Section 3, followed by examples in Section 4 and then conclusions.

2 CUTTING SURFACE PARAMETERIZATION

First, an initial design is chosen and mapped onto a design domain using an implicit signed-distance function, where the boundary is the zero level-set and a positive value indicates a point inside the structure, Figure 1. The structure is then defined by the scalar value of an implicit function:

$$\begin{cases} \phi(x) < 0, & x \notin \Omega \\ \phi(x) = 0, & x \in \Gamma \\ \phi(x) \geq 0, & x \in \Omega \end{cases} \quad (1)$$

where $\phi(x)$ is the implicit function, x is a point in the design domain (Ω_D), Ω and Γ are the structural domain and boundary, respectively.

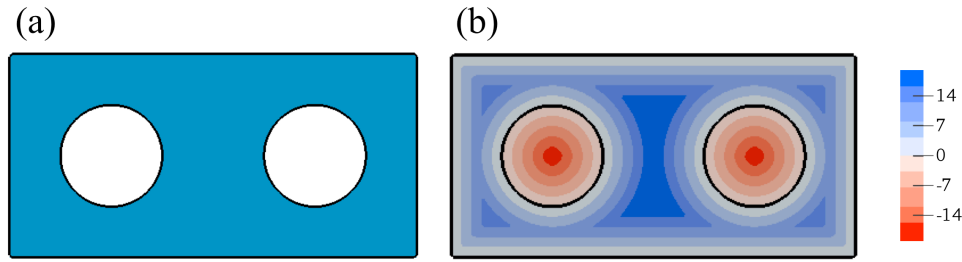


Figure 1. a) Original structure, b) Contours of the signed distance function.

Another way to interpret this representation of the structure is that the boundary is the zero-level set of a higher dimensional implicit function. Therefore, we can say that the boundary is the intersection of a plane (defined by all points where $\phi = 0$) and the implicit function surface in the higher dimension. This is shown in Figure 2a for the 2D structure in Figure 1. To change the position of the structural boundary we can fix the position of the intersecting plane and change the implicit function, which is approach in most level-set topology optimization methods [5]. Alternatively, we could change the position of the intersecting plane. This second approach is demonstrated in Figure 2b, where the intersecting plane has been moved, resulting in the structural boundary moving position, as shown in Figure 2c.

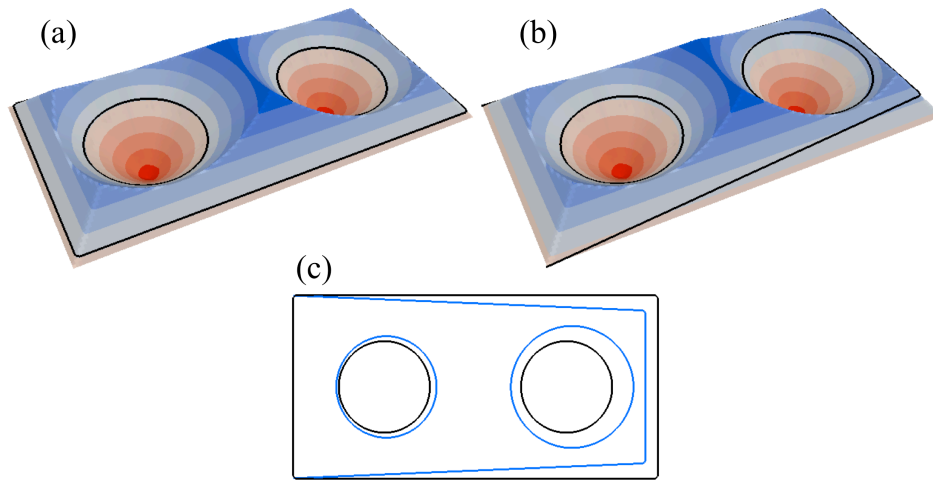


Figure 2. a) Implicit function surface with zero level-set contour, b) new position of the intersecting plane, resulting in a change of the structural boundary, c) comparison of structure boundaries: zero level-set (black) and new position of intersecting plane (blue).

The aim of this paper is to create a topology optimization method that exploits the moving intersecting plane mechanism for changing the position and connectivity of the structural boundary. The position of a plane in 3D can be described by 3 variables and we could proceed to develop an optimization method for 2D structures that uses just 3 variables. However, the design space would be very limited and thus this approach is not practical.

To provide more design freedom, the intersecting plane is replaced by a cutting surface that is defined in the higher dimension. Therefore, the method introduced in this paper uses two scalar valued functions: the implicit signed-distance function of the initial structure, $\alpha(x)$,

and the cutting surface, $\beta(x)$. The boundary of the structure is then defined by the set of points where the value of the signed-distance function equals the value of the cutting surface. Alternatively, the structural boundary is defined as the zero level-set of the implicit scalar-valued function that is the sum of the signed-distance and cutting surface functions. Thus, the definition of $\phi(x)$ in Eq. (1) can be written as:

$$\phi(x) = \alpha(x) + \beta(x) \quad (2)$$

The definition of the cutting surface is parameterized and the parameters become the design variables during optimization. There are several possible methods to parameterize the cutting surface, such as radial basis functions, polynomials or spline functions. In this paper, the cutting surface is parameterized using finite element shape functions, which is detailed in the next section.

During optimization, the signed-distance function remains fixed and the structure boundary changes position (and possibly connectivity) as the parameters of the cutting surface change. However, the design space is then limited by the current signed-distance function. Therefore, once an optimal cutting surface has been found for the current signed-distance function, a new signed-distance function is generated from the boundary of the current structure. The optimal cutting surface for the new signed-distance function is then found. This leads to an optimization method with an inner and outer loop. The aim of inner loop is to find the optimal cutting surface for the current signed-distance function. The outer loop then generates a new signed-distance function from the new structural boundary (as defined by the old signed-distance function and optimal cutting surface). At the start of the inner loop, the cutting surface is initialized as: $\beta(x) = 0$ (identical to the zero level-set), so that the structure boundary is the same as the optimum found by the previous inner loop (or initial structure).

In this paper, the inner loop optimization problem is solved by efficient gradient-based mathematical programming methods. Therefore, at least the first order derivatives of the objective and constraint functions need to be computed. It is usually most efficient to use analytical derivatives. Thus, an expression for the derivatives in terms of the cutting surface parameters is required. Structural quantities of interest can often be stated as integrals of a functional over the design domain, for example, compliance:

$$C(u, \phi(\alpha, \beta(\mathbf{b}))) = \int_{\Omega_D} c(u, x) \cdot H(\phi(x, \alpha, \beta(\mathbf{b}))) dx \quad (3)$$

where $C(u, \phi)$ is the total compliance of the structure, u is the displacement vector, $c(u, x)$ is the dot product of the stress and strain tensors at point x , \mathbf{b} is a vector of the cutting surface parameters (the design variables) and $H(\phi)$ is the Heaviside function:

$$H(\phi) = \begin{cases} 1 & \phi \geq 0 \\ 0 & \phi < 0 \end{cases} \quad (4)$$

To compute the integral in Eq. (3) numerically the design domain is usually discretized using finite elements:

$$C(u, \phi(\alpha, \beta(\mathbf{b}))) = \sum_{e=1}^N \sum_{g=1}^{M_e} c(u_{e,g}) \cdot H(\phi(\alpha_{e,g}, \beta_{e,g}(\mathbf{b}))) \cdot w_{e,g} \quad (5)$$

where N is the total number of elements, M_e is the number of integration points for element e and $w_{e,g}$ is an integration point weight. Taking the derivative of Eq. (5) with respect to design variable b_i , gives:

$$\frac{dC}{db_i} = \sum_{e=1}^N \sum_{g=1}^{M_e} \left[\frac{\partial c(u_{e,g})}{\partial u_{e,g}} \cdot \frac{\partial u_{e,g}}{\partial b_i} \cdot H(\phi_{e,g}) + c(u_{e,g}) \cdot \frac{\partial H(\phi_{e,g})}{\partial \phi_{e,g}} \cdot \frac{\partial \phi_{e,g}(\beta_{e,g})}{\partial \beta_{e,g}} \cdot \frac{\partial \beta_{e,g}}{\partial b_i} \right] \cdot w_{e,g} \quad (6)$$

Therefore, the derivative of the compliance function (and many functions of interest in structural design) requires the derivative of the Heaviside function with respect to the implicit function, which is the Dirac delta function:

$$\frac{\partial H(\phi)}{\partial \phi} = \delta(\phi) = \begin{cases} 1 & \phi = 0 \\ 0 & \text{else} \end{cases} \quad (7)$$

Therefore, the derivative in Eq. (6) is discontinuous and only non-zero at the boundary of the structure (where $\phi = 0$). Thus, only integration points that lie exactly on the boundary will contribute to the derivative. The discontinuous nature of the derivatives using the exact Heaviside approach, Eq. (7), makes it difficult for gradient-based optimizers to find a solution. To avoid this problem, the Heaviside function can be approximated with a smooth function and this is the approach used in this paper. The choice of smoothed Heaviside function and inner loop optimization are discussed in Section 3.

3 NUMERICAL IMPLEMENTATION

3.1 Cutting surface parameterization

The proposed method uses two meshes that cover the design domain: one to discretize the cutting surface (cutting surface mesh) and one to perform the finite element analysis (analysis mesh), as shown in Figure 3.

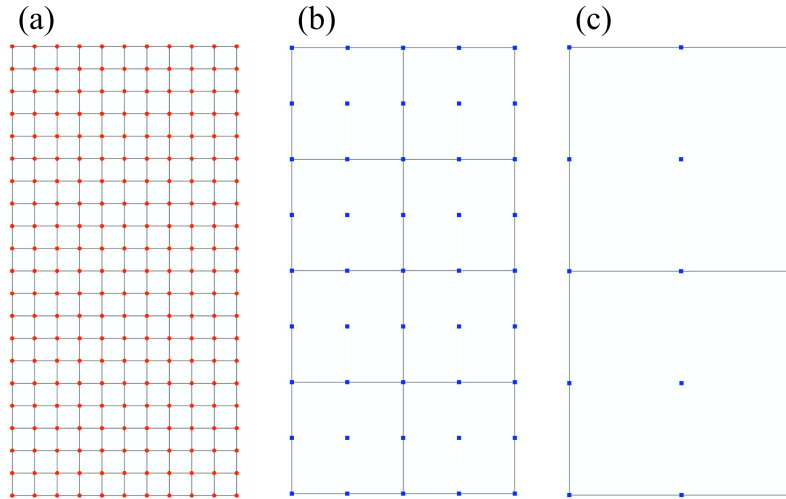


Figure 3. a) Analysis mesh using 200 Q4 elements, b) Cutting surface mesh using Q9 elements and mesh ratio 5 (45 design variables), c) cutting surface mesh with mesh ratio 10 (15 design variables).

The value of $\beta(x)$ is defined at nodes in the cutting surface mesh and interpolated using polynomial shape functions. The nodal values of $\beta(x)$ are then the design variables \mathbf{b} . The implicit function, ϕ , and signed-distance function, α , are discretized on the analysis mesh. Values of ϕ at analysis element integration points are obtained by interpolating ϕ using the analysis element shape functions. Equation (2) can be written in terms of the discretized variables:

$$\boldsymbol{\phi} = \boldsymbol{\alpha} + \mathbf{A} \cdot \mathbf{b} \quad (8)$$

where $\boldsymbol{\phi}$ and $\boldsymbol{\alpha}$ are vectors of the discretized implicit and signed-distance functions, respectively, and \mathbf{A} is a coefficient matrix that maps the cutting surface values to the nodes of the analysis mesh. This matrix remains constant during optimization and is only computed once.

Typically, the analysis and cutting surface meshes are different and there are advantages to decoupling the analysis discretization and design parameterization, such as reducing the number of design variables [16]. If the same mesh is used for both analysis and cutting surface, then \mathbf{A} is the identity matrix and the method resembles one using nodal design variables. To reduce the number of design variables, the cutting surface mesh uses fewer elements than the analysis mesh. To aid future discussion, the ratio of analysis elements to cutting surface elements in one direction is called the “mesh ratio.”

In this paper, 4 node bilinear elements (Q4) are used to discretize the analysis mesh and second order polynomial elements with 9 terms (Q9) are used to discretize the cutting surface, Figure 3. Thus, in this case, a mesh ratio of 2 results in \mathbf{A} becoming the identity matrix.

Finally, the signed-distance function is computed before the start of each inner loop using the fast marching method developed by Adalsteinsson and Sethian [17].

3.2 Smoothed Heaviside function

In this paper a polynomial function is used to approximate the Heaviside function [026]:

$$H(\phi) = \begin{cases} \rho_{\min} & , \phi < -\Delta \\ \frac{3(1-\rho_{\min})}{4} \left(\frac{\phi}{\Delta} - \frac{\phi^3}{3\Delta^3} \right) + \frac{1+\rho_{\min}}{2} & , -\Delta \leq \phi < \Delta \\ 1 & , \phi \geq \Delta \end{cases} \quad (9)$$

where Δ is the smoothing length of the approximation and ρ_{\min} is a parameter that defines the stiffness of the void region, in a similar way to the minimum density value in an element-based topology optimization method.

One of the benefits of using an implicit function for topology optimization, compared with element-based methods, is that the solution has smooth, well-defined boundaries. However, this is partially lost when using a smoothed Heaviside function, as the boundary becomes effectively blurred over the smoothing length [5]. To counter this, two strategies are proposed. Firstly a small smoothing length is used throughout the optimization, ideally smaller than the element edge length in the analysis mesh. However, this strategy may cause convergence problems with some examples, especially those that have a small volume ratio. Thus, a second strategy is proposed where the optimization starts with a large smoothing length that is then reduced each time the outer loop converges. The optimization continues until a converged solution using a lower limit of the smoothing length is obtained. This strategy is similar to continuation techniques used in density-based element methods to reduce the number of grey elements [3].

3.3 Inner loop optimization problem

The inner loop optimization problem can be stated as:

$$\begin{aligned} & \text{Minimize : } f(\boldsymbol{\phi}(\mathbf{b})) \\ & \text{Subject to : } g_i(\boldsymbol{\phi}(\mathbf{b})) \leq 0 \quad , i = 1 \cdots m \\ & \mathbf{b}_{\min} \leq \mathbf{b} \leq \mathbf{b}_{\max} \end{aligned} \quad (10)$$

where $f(\phi)$ is the objective function, $g_i(\phi)$ the constraint functions, m the total number of constraints, \mathbf{b}_{\min} and \mathbf{b}_{\max} are the side limits on the design variables.

The solution to the inner loop problem is affected by the choice of optimizer, design variable side limits and convergence criteria. In this paper the inner loop problem is solved using the method of moving asymptotes, as implemented in the package NLOPT [18]. The side limits are defined from the maximum and minimum signed-distance values of the current structure, with considerations for the boundary of the design domain, Γ_D , and fixed, non-designable regions within the design domain, Ω_{fixed} . First, a parameter is obtained from the discrete values of the signed-distance function:

$$\hat{\alpha} = \max\left\{\min\left\{\max\{\alpha_i\}, -\min\{\alpha_i\}\right\}, 2h\right\} \quad (11)$$

The side limits for a design variable are then defined as:

$$b_{i,\min} = \begin{cases} 0 & \text{if } b_i \in \Omega_{\text{fixed}} \\ -\hat{\alpha} & \text{else} \end{cases} \quad b_{i,\max} = \begin{cases} -\phi_i & \text{if } b_i \in \Gamma_D \\ \hat{\alpha} & \text{else} \end{cases} \quad (12)$$

Three convergence criteria are used for the inner loop: the maximum number of iterations, the relative change of the objective function and relative change of the design variables. The relative change criteria are both set to 10^{-3} and the maximum number of iterations to 10. The inner loop stops when any of these criteria are met. The criteria are intentionally slack, as this reduces the number of inner loop iterations spent fine tuning the structure, whereas quicker progress is often made by obtaining an approximate solution and generating a new signed-distance function for the next inner loop.

4 EXAMPLES

4.1 Minimization of compliance

First, the method introduced in this paper is used to solve two classic problems. The objective is to minimize compliance, with an upper limit on the total volume. The first problem is a cantilever, ratio 2:1, shown in Figure 4a, with volume constraint set to 50% of the design domain. The second is a Michell arch, shown in Figure 4b, with a volume constraint of 40%. The grey areas show fixed regions within the design domain. For both examples, the material properties are Young's modulus 1.0 and Poisson's ratio 0.3, and the load magnitude is 10 units. The initial design for both examples is shown in Figure 4c. The parameters for the smoothed Heaviside, Eq. 9, are: $\Delta = 0.25h$ and $\rho_{\min} = 10^{-6}$. Both examples use an analysis mesh composed of 160×80 unit sized plane stress elements. They are solved using different cutting surface meshes, defined by mesh ratios: 5, 10 and 20. This corresponds to problems with: 2145, 561 and 153 design variables, respectively.

The solutions for the cantilever are shown in Figure 5 and the solutions for the Michell arch are shown in Figure 6. The topologies and general shapes of the solutions compare well with the known analytical solutions and the solutions obtained using other methods [3, 8, 19]. For the cantilever, the solution using a mesh ratio of 5 is more topologically complex than the other solutions that use fewer cutting mesh elements. The compliance values for the three solutions are 5985, 6040 and 6131, for mesh ratios 5, 10 and 20, respectively. Thus, the objective value increases with a decrease in the number of design variables (or number of cutting surface elements). In this case using 153 design variables returns a compliance value 2.4% higher than the solution obtained using 2145 design variables. Also, for the mesh ratio 20 so-

lution, some parts of the structure boundary are not as straight as the boundaries of the other two solutions.

For the Michell arch, the topology of the main structural members is the same for all three solutions. The compliance values are 1095, 1090 and 1128, for mesh ratios 5, 10 and 20, respectively. Thus, there is a 3.5% increase in compliance for the mesh ratio 20 solution, compared with the mesh ratio 10 solution. Again, parts of the boundary for the mesh ratio 20 solution are not straight, which contributes to the increase in compliance.

Overall, these two examples demonstrate that the proposed method can obtain solutions to well-known compliance minimization problems. Also, there is a limit to how far the design space can be reduced (by reducing the number of cutting mesh elements), before the objective value and quality of the boundary become significantly affected. Note that, for these examples, a mesh ratio of 10 uses 23 times less design variables than the number of analysis mesh elements.

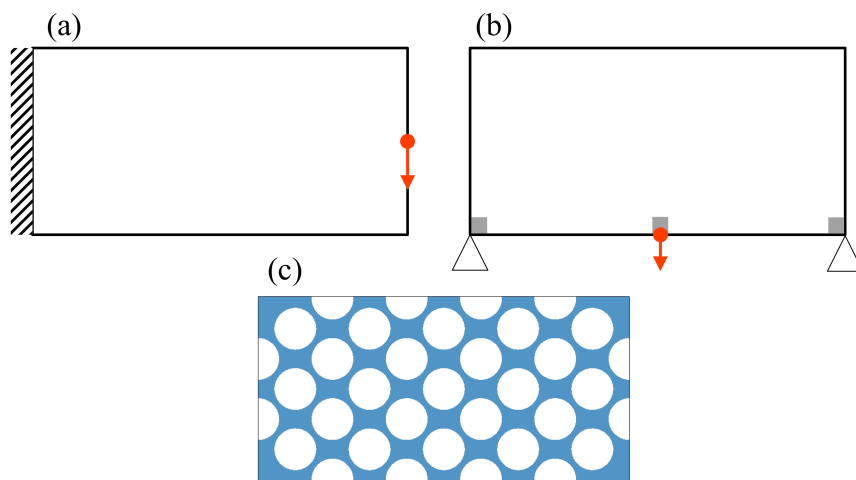


Figure 4. Examples. a) Cantilever, b) Michell arch, c) Initial design.



Figure 5. Cantilever minimization of compliance solutions. a) Mesh ratio 5, b) mesh ratio 10, c) mesh ratio 20.



Figure 6. a) Michell arch minimization of compliance solutions a) Mesh ratio 5, b) mesh ratio 10, c) mesh ratio 20.

4.2 Minimization of volume

The two examples used in the previous section for compliance minimization and now solved for the complimentary problem: minimization of volume, subject to an upper limit on compliance. For the cantilever, the upper limit is 6000 units and for the Michell arch the limit is 1100 units. These limits are approximately the optimal compliance values obtained in the previous section. Again, both problems are solved using different cutting surface meshes, defined by mesh ratios: 5, 10 and 20. All other parameters are the same as the previous section.

The solutions for the cantilever are shown in Figure 7 and the solutions for the Michell arch are shown in Figure 8. In all cases, a feasible solution is obtained. For the cantilever, the final volume values are 6353, 6454 and 6838, for mesh ratios 5, 10 and 20, respectively. Thus, there is a 7.6% increase in volume for the mesh ratio 20 solution, compared with the mesh ratio 5 solution. A similar trend is observed for the Michell arch, where the volume values are 5085, 5313 and 5597 for mesh ratios 5, 10 and 20, respectively. In this case the volume increase is 10.1%. The percentage increase in objective value is higher than that observed for the compliance minimization problems. This is perhaps because the compliance constraint is nonlinear and thus poses a more difficult problem to solve than one with a linear volume constraint. Again, the quality and definition of the boundary decreases as the number of design variables decreases.

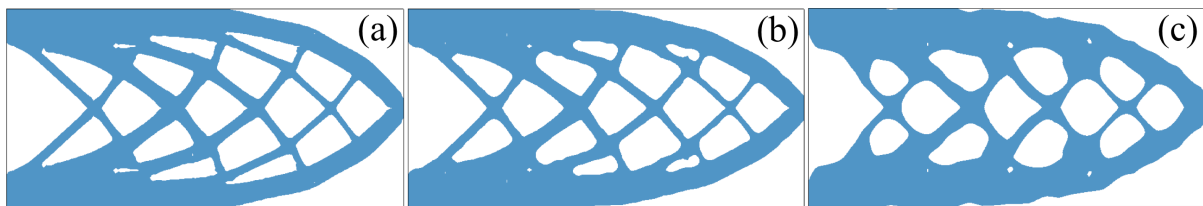


Figure 7. Cantilever minimization of volume solutions. a) Mesh ratio 5, b) mesh ratio 10, c) mesh ratio 20.



Figure 8. a) Michell arch minimization of volume solutions a) Mesh ratio 5, b) mesh ratio 10, c) mesh ratio 20.

4.3 Compliant mechanism

The new method is now used to solve an inverter compliant mechanism problem [14], as shown in Figure 9a. The objective is to maximize the output displacement, which is defined as positive in the opposite direction to the input force, with an upper limit on the total volume, set to 15% of the design domain. The material properties are Young's modulus 100.0 and Poisson's ratio 0.3, and the input load magnitude is 10 units. The stiffness of the input and output springs are: $k_{in} = 0.001$, $k_{out} = 1.0$. The initial design is shown in Figure 9b. The design domain is discretized using 160×160 plane stress elements, with element edge length 0.5.

For this example, a continuation strategy is used to adapt the smoothing length during optimization (as discussed in Section 3.2). The initial value is $\Delta = 2h$ and the lower limit is

$0.25h$, where h is the analysis mesh grid spacing. The smoothing length is halved each time the outer loop converges. The value of $\rho_{\min} = 10^{-2}$.

The solutions for mesh ratios 5 and 10 (4225 and 1089 design variables, respectively) are shown in Figure 10. These correspond well with solutions for similar problems obtained using other methods [3, 20, 21]. The output displacement values are 1.37 and 1.25, for mesh ratios 5 and 10 respectively. Both solutions have the same topology, with small differences in shape. This is particularly noticeable near the input and output locations, where the mesh ratio 5 solution uses very thin structural members to create compliant hinges, whereas the hinges for the mesh ratio 10 solution are thicker. This difference results in the 9.8% increase in output displacement for mesh ratio 5, compared with mesh ratio 10. However, this increase is partly obtained by exploiting the numerical discretization, where a very compliant hinge is created around one node in the analysis mesh [22]. It would be interesting to add minimum length scale control to the proposed method to avoid this issue. The problem was also solved using a mesh ratio of 20. The solution was feasible (the volume constraint was met), but there was no physical link between the input and output locations, so it is not a practical design.

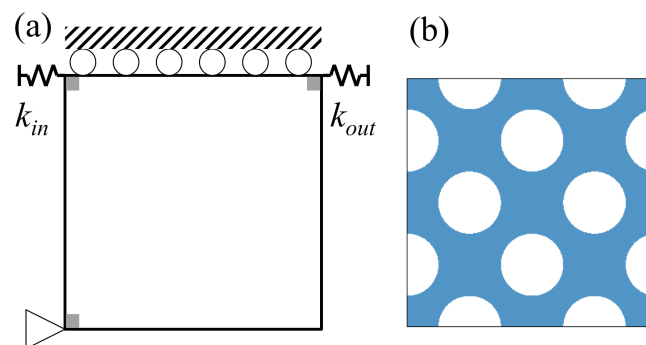


Figure 9. a) Inverter example, b) Inverter initial design.

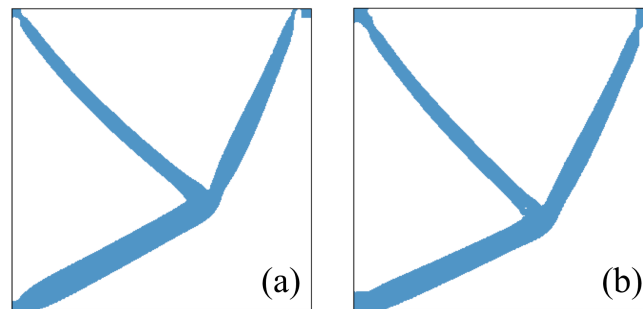


Figure 10. Inverter solutions. a) Mesh ratio 5, b) mesh ratio 10.

5 CONCLUSIONS

A new topology optimization method is proposed that uses an implicit signed-distance function and cutting surface, where the structure boundary is defined by their intersection. During optimization, the implicit function is fixed and the cutting surface modified to change the position and connectivity of the boundary. The cutting surface is parameterized using finite element shape functions and the nodal values become the design variables during optimization. The analysis and cutting surface meshes are different, where the cutting surface mesh uses fewer elements than the analysis mesh, thus reducing the number of design variables compared with a method using element-wise design variables. A critical aspect of the new method is the use of a smoothed Heaviside function for integrating the structural analysis ma-

trices. This approach avoids the discontinuous derivatives associated with an exact Heaviside approach. The benefits of the new method are that: solutions have clear, smooth boundaries, efficient mathematical programming methods can be employed and the number of design variables is reduced.

The new method is demonstrated by solving well-known minimization of compliance, minimization of volume and complaint mechanism problems. In each case, feasible, smooth boundary solutions are obtained that agree well with solutions obtained using other methods. However, there is a limit to how far the number of design variables can be reduced before the objective value and quality of the boundary become significantly affected. Despite this, good solutions to all problems are obtained using a number of design variables around 23 times fewer than the number of elements used for analysis.

REFERENCES

- [1] J.D. Deaton, R.V. Grandhi, A survey of structural and multidisciplinary continuum topology optimization: post 2000. *Structural and Multidisciplinary Optimization*, **49**, 1-38, 2014.
- [2] O. Sigmund, K. Maute, Topology optimization approaches: A comparative review. *Structural and Multidisciplinary Optimization*, **48**(6), 1031-1055, 2013.
- [3] M.P. Bendsoe, O. Sigmund, Topology optimization: Theory, Methods and Applications. Springer-Verlag, 2004.
- [4] Y.M. Xie, G.P. Steven, Evolutionary structural optimization. Springer, 1997.
- [5] N. P. van Dijk, K. Maute, M. Langelaar, F. van Keulen, Level-set methods for structural topology optimization: a review. *Structural and Multidisciplinary Optimization*, **48**, 437-472, 2013.
- [6] M.Y. Wang, X. Wang, D. Guo, A level set method for structural topology optimization. *Computer Methods In Applied Mechanics and Engineering*, **192**(1-2), 227-246, 2003.
- [7] G. Allaire, F. Jouve, A.M. Toader, Structural optimization using sensitivity analysis and a level-set method. *Journal of Computational Physics*, **194**(1), 363-393, 2004.
- [8] P.D. Dunning, H.A. Kim. Introducing the sequential linear programming level-set method for topology optimization. *Structural and Multidisciplinary Optimization*, **51**(3), 631-643, 2015.
- [9] S. Wang, M. Wang, Radial basis functions and level set method for structural topology optimization. *International Journal For Numerical Methods In Engineering*, **65**(12), 2060-2090, 2006.
- [10] W. Zhang, J. Zhang, X. Guo, Lagrangian description based topology optimization - a revival of shape optimization. *Journal of Applied Mechanics*, **83**(4), 2016.
- [11] A.A. Gomes, A. Suleman, Application of spectral level set methodology in topology optimization. *Structural and Multidisciplinary Optimization*, **31**(6), 430-443, 2006.
- [12] N.P. van Dijk, M. Langelaar, F. van Keulen, Explicit level-set-based topology optimization using an exact Heaviside function and consistent sensitivity analysis. *International Journal for Numerical Methods in Engineering*, **91**(1), 67-97, 2012.

- [13] J.K. Guest, Topology optimization with multiple phase projection. *Computer Methods in Applied Mechanics and Engineering*, **199**(1-4), 123-135, 2009.
- [14] S. Rojas-Labanda, M. Stolpe, Benchmarking optimization solvers for structural topology optimization. *Structural and Multidisciplinary Optimization*, **52**(3), 527-547, 2015.
- [15] S. Wang, E.D. Sturler, G.H. Paulino, Large-scale topology optimization using preconditioned Krylov subspace methods with recycling. *International journal for numerical methods in engineering*, **69**(12), 2441-2468, 2007.
- [16] J.K. Guest, L.C. Smith Genut, Reducing dimensionality in topology optimization using adaptive design variable fields. *International journal for numerical methods in engineering*, **81**: 1019–1045, 2010.
- [17] D. Adalsteinsson, J. A. Sethian, The fast construction of extension velocities in level set methods. *Journal of Computational Physics*, **148**(1), 2-22, 1999.
- [18] The NLOpt nonlinear optimization package. <http://ab-initio.mit.edu/nlopt>. 2014.
- [19] A.L. Gain, G.H. Paulino, A critical comparative assessment of differential equation-driven methods for structural topology optimization. *Structural and Multidisciplinary Optimization*, **48**(4), 685-710, 2013.
- [20] O. Sigmund, On the design of compliant mechanisms using topology optimization. *Mechanics of Structures and Machines*, **25**(4), 493-524, 1997.
- [21] Z. Luo, L. Tong, M.Y. Wang, S. Wang, Shape and topology optimization of compliant mechanisms using a parameterization level set method." *Journal of Computational Physics*, **227**(1), 680-705, 2007.
- [22] T.A. Poulsen, A simple scheme to prevent checkerboard patterns and one-node connected hinges in topology optimization. *Structural and Multidisciplinary Optimization*, **24**(5), 396-399, 2002.