Struct Multidisc Optim (2008) 35:89–92
DOI 10.1007/s00158-007-0140-4

BRIEF NOTE

A new look at ESO and BESO optimization methods

X. Huang · Y. M. Xie

Received: 14 March 2007 / Revised: 29 March 2007 / Accepted: 4 April 2007 / Published online: 17 May 2007
© Springer-Verlag 2007

Abstract The “hard-kill” optimization methods such as evolutionary structural optimization (ESO) and bidirectional evolutionary structural optimization (BESO) may result in a nonoptimal design (Zhou and Rozvany in Struct Multidisc Optim 21:80–83, 2001) when these methods are implemented and used inadequately. This note further examines this important problem and shows that failure of ESO may occur when a prescribed boundary support is broken for a statically indeterminate structure. When a boundary support is broken, the structural system could be completely changed from the one originally defined in the initial design and even BESO would not be able to rectify the nonoptimal design. To avoid this problem, it is imperative that the prescribed boundary conditions for the structure be checked and maintained at each iteration during the optimization process. Several simple procedures for solving this problem are suggested. The benchmark problem proposed by Zhou and Rozvany (Struct Multidisc Optim 21:80–83, 2001) is revisited, and it is shown that the highly nonoptimal design can be easily avoided.

Keywords “Hard-kill” optimization methods · Evolutionary structural optimization (ESO) · Bidirectional evolutionary structural optimization (BESO)

X. Huang (✉) · Y. M. Xie
School of Civil, Environmental and Chemical Engineering,
RMIT University,
GPO Box 2476V, Melbourne 3001, Australia
e-mail: huang.xiaodong@rmit.edu.au

Y. M. Xie
e-mail: mike.xie@rmit.edu.au

1 Introduction

Over the last three decades, many mathematical and heuristic optimization methods have been developed (Bendsøe and Kikuchi 1988; Zhou and Rozvany 1991; Xie and Steven 1993, 1997). Among them, evolutionary structural optimization (ESO) method uses the concept of gradually removing (“hard-kill”) redundant material from a structure so that the resultant structure evolves toward an optimum. Bidirectional evolutionary structural optimization (BESO) method (Yang *et al*. 1999) is an extension of ESO, which allows for inefficient materials to be removed from a structure at the same time as the efficient ones to be added. In so doing, the BESO method greatly improves the robustness of the solution process compared to traditional ESO method.

The topology optimization for a continuum structure is complex due to a large number of design variables. ESO offers an alternative method for solving various topology optimization problems of continuum structures (Chu et al. 1996; Li *et al*. 1999). Theoretically, it is noted that the sequential linear programming (SLP)-based approximate optimization method followed by the Simplex algorithm is equivalent to ESO/BESO (Tanskanen 2002). However, ESO-type methods have been questioned by Zhou and Rozvany (2001) who have demonstrated, correctly, a case where ESO could produce a nonoptimal design. The benchmark example of Zhou and Rozvany is shown in Fig. 1 where Young’s modulus is unity and Poisson’s ratio as zero. If the design domain is discretized using 100 four-node quadrilateral elements (Fig. 1b), a “hard-kill” method such as ESO would immediately produce a highly nonoptimal design (Fig. 1c) in the first iteration. Once the vertical bar is broken, the optimization process will be trying to solve a completely different structural system (and, of

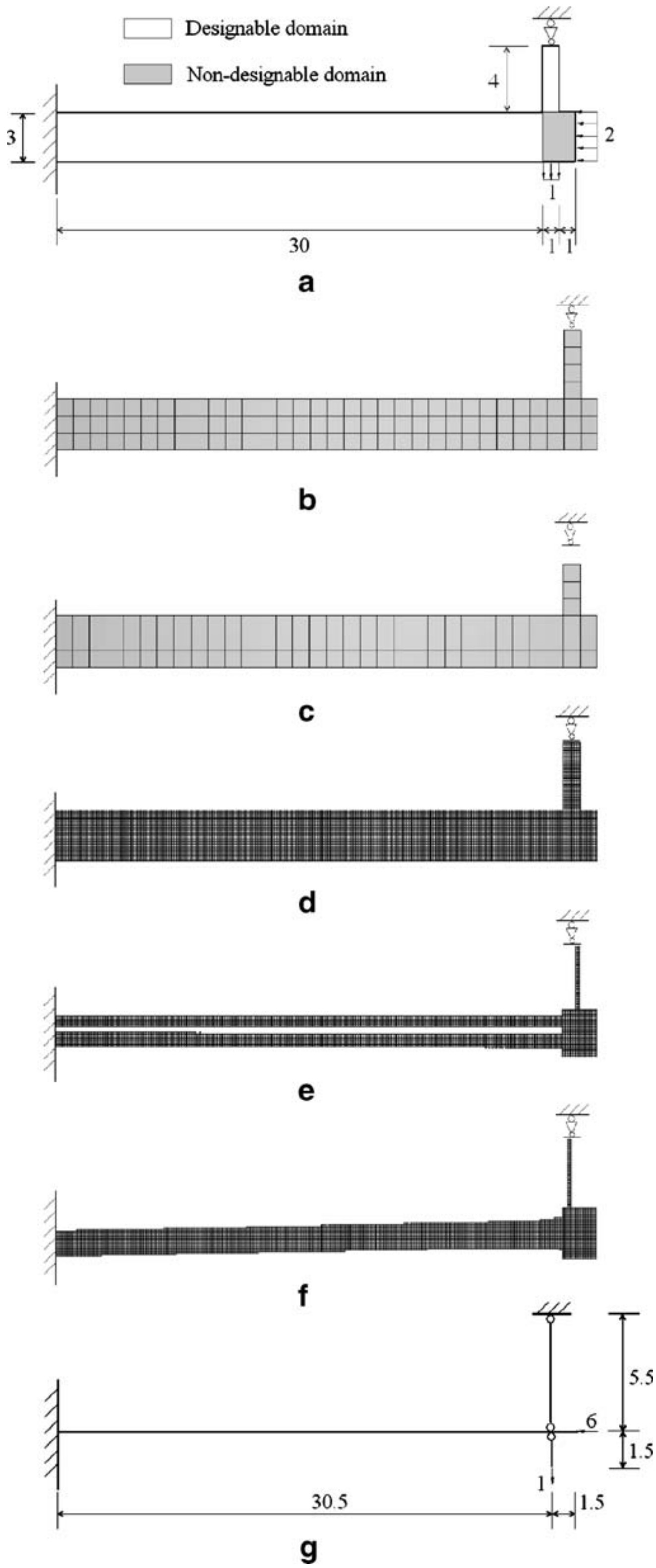

Fig. 1 **a** Design domain, load, and support conditions of example 1, **b** initial design with coarse mesh, **c** topology after iteration 1 with the breakdown of the roller support, **d** initial design with fine mesh, **e** ESO optimal design with C=380.5, **f** BESO optimal design with C=378.4, **g** simple beam-tie model for analytical verification (Zhou and Rozvany 2001)

course, resulting in a wrong solution), and even the BESO method would not be able to rectify this problem.

2 Method for detecting and rectifying the broken boundary support

In fact, the above problem commonly exists in the topology optimization of some statically indeterminate structures when "hard-kill" optimization methods are used. To overcome this problem, it is necessary to check the boundary conditions after each iteration for any "hard-kill" optimization method. A simple way is to define a group of nodes for each support. For example, in the above example, two node groups are defined for the fixed boundary on the left and roller support at the top, respectively. Once all nodes in a boundary group have no element connected, the optimization program should not proceed any further until some corrective measures are put in place. Note that when a boundary support is broken, the structural system could be completely changed from the one originally defined in the initial design. In the example of Zhou and Rozvany (2001), the breakdown of the support changes the structure into a cantilever beam, which is totally different from the original structural system.

The method for detecting the breakdown in boundary conditions as given above is easy to implement in computer code. Indeed, the FEA software we use, ABAQUS, has this method embedded in its solution procedure as part of its acceptable format of the input file.

Once the optimization process detects the breakdown of the prescribed boundaries, one could recover and then freeze the elements removed in the prior iteration. This technique is not recommended here due to the following two reasons. First, for a complex structure, a number of elements apart from those directly connected to the boundaries may be removed in one iteration, and therefore, it is difficult to determine which elements should be recovered. Second, the artificially frozen elements may result in an extremely inefficient support with very low-stress level.

Another way to avoid the problem is to replace the "hard-kill" approach with "soft-kill", which means to replace the removed elements with low-density elements. In the "soft-kill" optimization method, the prescribed boundaries are never removed. However, the final topology may oscillate between two solutions, and the selected design may have an extremely inefficient support as above. On the other hand, the "soft" elements in nonlinear structures may cause the tangent stiffness matrix to become indefinite or even negative definite (Burns and Tortorelli 2003).

Returning to Zhou and Rozvany (2001), example shown in Fig. 1a, it is seen that the vertical bar has a stress level approximately half of that in the horizontal bar. To optimize the structure, the vertical bar needs to be reduced roughly by half in size. This is impossible to achieve for any "hard-kill" method using the finite element mesh given in Fig. 1b because we cannot cut the elements in the vertical bar in halves. This is the mesh Zhou and Rozvany (2001) have used to "test" the ESO method and no wonder that it results in the "highly nonoptimal design" shown in Fig. 1c.

In the next section, we shall show that to solve this problem using ESO/BESO (or any other "hard-kill") method, a finer mesh has to be used.

3 ESO and BESO "optimal" solutions with a fine mesh

To minimizing the mean compliance $\left(\frac{1}{2}\mathbf{f}^T\mathbf{u}\right)$ of the structure, the design domain is discretized using 10,000 four-node elements as shown in Fig. 1d. In this example, we specify that the volume of the final design should be equal to 50% of that of the initial design. Of course, one could specify a different volume fraction for the final design.

When the rejection ratio is taken as 0.1% (i.e., ten elements are removed from the structure at each iteration), the ESO method obtains the final design shown in Fig. 1e, which satisfies the volume constraint. The mean compliance of the design is 380.5 from finite element analysis. For the BESO method, a higher rejection ratio can be used without reducing the accuracy of the solution. In the present study, the rejection ratio is selected to be 0.2% (i.e., a net of 20 elements are removed at each iteration). Its final design in Fig. 1f has the mean compliance of 378.4. In both optimization processes, the boundary conditions are checked after each iteration using the method described in Section 2. During both optimization processes, the breakdown of boundary support has never occurred. The structural performance of BESO design is slightly better than that of ESO design, although their topologies look different. Because ESO cannot recover any material previously removed, it is expected that the solution of BESO is usually better than that of ESO.

If one does detect a breakdown of boundary support before a satisfactory solution is obtained, it may well indicate that the mesh is still not fine enough for "hard-kill" optimization, and further refinement is required to obtain a correct solution.

A simple beam-tie model proposed by Zhou and Rozvany (2001) offers some useful insight into the ESO/BESO solutions. For the simple beam-tie model shown in

Fig. 1g, obviously, the optimal solution is the one where the width ratio between the vertical and horizontal bars is 1:6. To satisfy the objective volume fraction of 50%, the widths of the vertical and horizontal bars should be about 0.25 and 1.5, respectively. Thus, the mean compliance of the system is

$$C = \frac{1}{2}\left(\frac{7}{0.25} + 32 \times \frac{6}{1.5} \times 6\right) = 387.5$$

It can be seen that the mean compliances of ESO and BESO designs are very close to the simplified theoretical result. Therefore, both designs can be considered to be "optimal" solutions from the perspective of engineering applications, although one could still argue the designs are "suboptimal" theoretically. The theoretical bound on the structural performance can be obtained by a so-called microstructure or homogenization based approach (Bendsøe and Kikuchi 1988). However, the resulting design cannot be built since no definite length-scale is associated with the microstructure.

4 Conclusions

This note revisits the optimization problem presented in Zhou and Rozvany (2001) to further examine ESO type optimization methods. It shows that any breakdown of a prescribed boundary support should not be allowed in the "hard-kill" optimization process; otherwise, it may result in a highly nonoptimal design. Simple methods for detecting and rectifying the problem have been proposed. It is also demonstrated that ESO and BESO can obtain "optimal" designs for the test example if an adequate mesh is assigned. In general, if a breakdown of boundary support is detected before a satisfactory solution is obtained, it may well indicate that the used mesh is too coarse for "hard-kill" optimization, and a refinement is required to obtain a correct solution. It is worth noting that a parallel but independent investigation on the problem described in this paper has been carried out by Edwards et al. (2007).

References

Bendsøe MP, Kikuchi N (1988) Generating optimal topologies in structural design using a homogenization method. Comput Methods Appl Mech Eng 71:197–224

Burns TE, Tortorelli DA (2003) An element removal and reintroduction strategy for the topology optimization of structures and compliant mechanisms. Int J Numer Methods Eng 57:1413–1430

Chu DN, Xie YM, Hira A, Steven GP (1996) Evolutionary structural optimization for problems with stiffness constraints. Finite Elem Anal Des 21:239–251

Edwards CS, Kim HA, Budd CJ (2007) An evaluative study on ESO and SIMP for optimizing a cantilever tie-beam. Struct Multidisc Optim (in press)

Li Q, Steven GP, Xie YM (1999) Shape and topology design for heat conduction by evolutionary structural optimization. Int J Heat Mass Transfer 42:3361–3371

Tanskanen P (2002) The evolutionary structural optimization method: theoretical aspects. Comput Methods Appl Mech Eng 191:5485–5498

Xie YM, Steven GP (1993) A simple evolutionary procedure for structural optimization. Comput Struct 49:885–896

Xie YM, Steven GP (1997) Evolutionary structural optimization. Springer, Berlin

Yang XY, Xie YM, Steven GP, Querin OM (1999) Bidirectional evolutionary method for stiffness optimization. AIAA J 37 (11):1483–1488

Zhou M, Rozvany GIN (1991) The COC algorithm, part II: topological, geometry and generalized shape optimization. Comput Methods Appl Mech Eng 89:197–224

Zhou M, Rozvany GIN (2001) On the validity of ESO type methods in topology optimization. Struct Multidisc Optim 21:80–83