Struct Multidisc Optim (2010) 41:671–683 DOI 10.1007/s00158-010-0487-9

FORUM DISCUSSION

**A further review of ESO type methods for topology optimization**

# Xiaodong Huang · Yi-Min Xie

Received: 22 July 2008 / Revised: 11 November 2009 / Accepted: 27 January 2010 / Published online: 6 March 2010

§c Springer-Verlag 2010

**Abstract** Evolutionary Structural Optimization (ESO) and its later version bi-directional ESO (BESO) have gained widespread popularity among researchers in structural opti- mization and practitioners in engineering and architecture. However, there have also been many critical comments on various aspects of ESO/BESO. To address those criticisms, we have carried out extensive work to improve the original ESO/BESO algorithms in recent years. This paper summa- rizes latest developments in BESO for stiffness optimization problems and compares BESO with other well-established optimization methods. Through a series of numerical exam- ples, this paper provides answers to those critical comments and shows the validity and effectiveness of the evolutionary structural optimization method.

**Keywords** Evolutionary Structural Optimization (ESO) ·

Bi-directional ESO (BESO) · Local optimum ·

Optimal design · Displacement constraint

# Introduction

Evolutionary structural optimization (ESO) method was firstly introduced by Xie and Stev[en](#_bookmark34) ([1992](#_bookmark34), [1993](#_bookmark35), [1997](#_bookmark36)). The idea is based on a simple and empirical concept that a structure evolves towards an optimum by slowly removing (hard-killing) elements with lowest stresses. To maximize

1. Huang (B) 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](mailto:huang.xiaodong@rmit.edu.au)

1. M. Xie

e-mail: [mike](mailto:mike.xie@rmit.edu.au)[.xie@rmit.edu.au](mailto:.xie@rmit.edu.au)

the stiffness of the structure, stress criterion was replaced with elemental strain energy criterion by Chu et a[l.](#_bookmark16) ([1996](#_bookmark16)). Bi-directional evolutionary structural optimization (BESO) (Querin et a[l.](#_bookmark25) [1998](#_bookmark25), [2000](#_bookmark26)) method is an extension of that idea which allows for new elements to be added in the locations next to those elements with highest stresses. For stiffness optimization problems using the strain energy cri- terion, Yang et [al.](#_bookmark37) ([1999](#_bookmark37)) estimated the strain energy of void elements by linearly extrapolating the displacement field.

ESO/BESO has been used for a wide range of applica- tions and hundreds of publications have been produced by researchers around the world. Several landmark buildings designed using ESO/BESO have now been constructed in Japan and Qatar (Cui et a[l.](#_bookmark17) [2005](#_bookmark17); Ohmori et a[l.](#_bookmark24) [2005](#_bookmark24)). Mean- while, some shortcomings have been pointed out by Sig- mund and P[etersson](#_bookmark32) ([1998](#_bookmark32)), Zhou and Rozvan[y](#_bookmark39) ([2001](#_bookmark39)) and Rozvan[y](#_bookmark27) ([2009](#_bookmark27)). Firstly, the original ESO/BESO methods fail to achieve a convergent optimal solution. As a result, we have to select the best solution by comparing a large number of solutions generated during the optimization process (Rozvan[y](#_bookmark27) [2009](#_bookmark27)). Secondly, a notable question about ESO/ BESO has arisen following the work of Zhou and Rozvany ([2001](#_bookmark39)) in which a highly inefficient solution to a cantilever tie-beam structure by the ESO method was pointed out. Thirdly, the ESO/BESO procedure cannot be easily extend- ed to other constraints such as displacement (Sigmund and [Petersson](#_bookmark32) [1998](#_bookmark32); Rozvan[y](#_bookmark27) [2009](#_bookmark27)).

In order to answer these critical comments, this paper is organized as follows. In Section [2](#_bookmark0), we briefly sum- marize the recent improvements in the BESO method. In Section [3](#_bookmark2), we compare the results of the BESO method with those from other optimization approaches. In Section [4](#_bookmark7), we revisit the cantilever tie-beam example in Zhou and Rozvan[y](#_bookmark39) ([2001](#_bookmark39)) and explore the essence of the inefficient solution. In Section [5](#_bookmark9), we extend the BESO method to an

optimization problem with a displacement constraint rather than a volume constraint which has been commonly used in the original ESO/BESO methods.

*i*

# Improvements in evolutionary structural optimization

where **K**0 denotes the elemental stiffness matrix for solid elements. In ESO/BESO methods, a structure is optimized using discrete design variables. That is to say that only two bound materials are allowed in the design. Hence, the rela- tive ranking of elemental sensitivities for both solid and soft elements can be expressed by

2.1 Problem statement and material interpolation scheme

⎧⎪⎪ 1 **u***T* **K**0**u***i* when *xi* = 1

1 *∂C*

⎨ 2 *i i*

In the original ESO and BESO method, no clear or com- pleted statements of the optimization problem such as objec-

*αi* =− *p ∂xi* =

*p*−1

min **u***T* **K**0**u***i* when *xi* = *x*min

*ρ*⎪⎪⎩

2

*i*

*i*

(4)

tive functions and constraints were presented. In the current

BESO method (Huang and X[ie](#_bookmark21) [2009a](#_bookmark21)), the optimization problem can be clearly stated as follow

1

=

Minimize: *C* **u***T* **Ku**

2

*N*

.

where *αi* is termed as the sensitivity number for the *i* th ele- ment. It is noted that the sensitivity numbers of soft elements depend on the selection of the penalty exponent *p*. When the penalty exponent tends to infinity, the above sensitivity number becomes

Subject to: *V* ∗ − *Vi xi* = 0

*i* =1

(1)

⎧⎪⎨ 1 *T* 0

*xi* = *x*min or 1

. Σ

Note that *C* is the mean compliance 1 **fTu** rather than the compliance .**fTu**Σ used in the SIMP method. **K** and **u** are the

2

*αi* =

2 **u***i* **K***i* **u***i* when *xi* = 1

⎪⎩0 when *xi* = *x*min

(5)

global stiffness matrix of the structure and the displacement vector. *Vi* is the volume of an individual element and *V* ∗ is the prescribed structural volume. *N* is the total number of elements. The binary design variable *xi* denotes the density of the *i* th element. It should be noted that a small value of *x*min e.g. 0.001 is used to denote the void elements.

In the original ESO/BESO method, the complete removal of a solid element from the design domain could result in theoretical difficulties in topology optimization. It appears to be rather irrational when the design variable (an element) is directly eliminated from the topology optimization prob- lem. Rozvany and [Querin](#_bookmark28) ([2002](#_bookmark28)) suggest a sequential ele- ment rejection and admission (SERA) method in which the void element is replaced by a soft element with a very low density. Huang and X[ie](#_bookmark21) ([2009a](#_bookmark21)) developed the BESO method utilizing the SIMP model where the material inter- polation scheme can be expressed by

This equation indicates that the sensitivity numbers of solid elements and soft elements are equal to the elemental strain energy and zero, respectively. According to the above mate- rial interpolation scheme, the Young’s modulus of soft elements also becomes zero as *p* approaches infinity. Due to the above reasons, when *p* tends to infinity *the soft elements are equivalent to void elements* and can be com- pletely removed from the design domain as in the original hard-kill ESO method (Chu et [al.](#_bookmark16) [1996](#_bookmark16)). Therefore, it is concluded that the hard-kill ESO method is a special case of the soft-kill BESO method where the penalty exponent *p* approaches infinity.

2.3 Mesh-independent filter and stabilization of evolutionary process

*E (x )* = *E x*

*p*

*i*

1

*i*

(2)

Basically, the BESO mesh-independency filter works sim-

where *E*1 denotes the Young’s modulus for solid material and *p* is the penalty exponent.

ilarly to that used in the SIMP method to avoid numerical

2.2 Sensitivity analysis and sensitivity number

Using the adjoint method, the sensitivity of the objective function with regard to the change in the *i* th element can be found as (Bendsøe and S[igmund](#_bookmark15) [2003](#_bookmark15))

*∂C px p*−1

=− *i* **u***T* **K**0**u***i* (3)

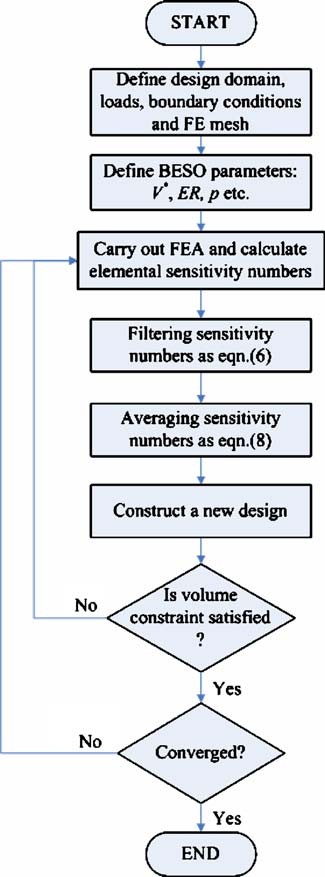
*i*

instabilities such as checkerboard and mesh-dependency. First, the nodal sensitivity numbers *(αn)* which do not carry any physical meaning on their own are defined by averaging the sensitivity numbers of connected elements. Then, these nodal sensitivity numbers must be converted back into ele- ment before the topology can be determined. Here, a filter scheme is used to carry out this process. The defined filter functions are based on a length scale *r*min. The primary role of the scale parameter *r*min in the filter scheme is to identify the nodes that influence the sensitivity of the *i* th element.

*∂xi*

1. *i i*

This can be visualized by drawing a circle of radius *r*min

centred at the centroid of *i* th element, thus generating the circular sub-domain *Ki* . Nodes located inside *Ki* contribute to the computation of the modified sensitivity number of the *i* th element as

*α*ˆ *i* =

*M*

*n j*

. *w* .*rij* Σ *α*

*j* =1

*M*

. . Σ

*w rij*

*j* =1

(6)

where *M* is the total number of nodes in the sub-domain *Ki* and *rij* denotes the distance between the center of the element *i* and node *j* . *w(rij )* is a weight factor given as

*w(r*

*r*min *rij* for *rij < r*min

*ij*

= .

*)* −

0 for *rij* ≥ *r*min

(7)

It can be seen that the above filter provides the sensitivity numbers for void elements through filtering the sensitiv- ity numbers of their neighboring solid elements. Hence, the hard-kill BESO method can be developed with the help of the mesh-independency filter (Huang and [Xie](#_bookmark19) [2007](#_bookmark19)).

However, the objective function and the corresponding topology may not be convergent because the sensitivity numbers are calculated based on the different status of ele- ments (1 or *x*min). Computational experience has shown that averaging the sensitivity number with its historical informa- tion is an effective way to avoid this problem (Huang and [Xie](#_bookmark19) [2007](#_bookmark19), [2009a](#_bookmark21)). It can be simply expressed by

*α*˜ *i* = 2 .*α*ˆ *i,k* + *α*ˆ *i,k*−1Σ (8)

1

where *k* is the current iteration number. Then let *αi,k αi* , thus the modified sensitivity number considers the sensi- tivity information in the previous iterations.

ˆ = ˜

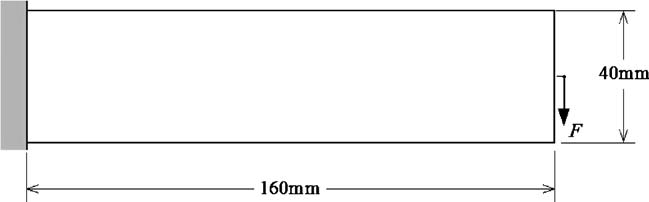
The optimality criteria for the stiffness optimization problem can be easily derived if no restriction is imposed on the design variables *xi* , i.e. the strain energy densities of all elements should be equal. Thus, the elements with higher strain energy density should have *xi* increased and the elements with lower strain energy density should have *xi* decreased. For the soft-kill BESO method, as the design variables are restricted to be either *x*min or 1, the optimality criteria can be described as that *strain energy densities of solid elements are always higher than those of soft elements*. Therefore, we devise an update scheme for the design vari- ables *xi* by changing from 1 to *x*min for elements with lowest sensitivity numbers and from *x*min to 1 for elements with highest sensitivity numbers. The threshold of sensitivity number can be easily determined by the target volume for

**Fig. 1** Flowchart of the BESO method

the next iteration and relative ranking of sensitivity numbers (Huang and X[ie](#_bookmark21) [2009a](#_bookmark21)). A flowchart of the BESO method is given in Fig. [1](#_bookmark1) and a simplified soft-kill BESO program written in Matlab code is given in the [Appendix](#_bookmark13).

# Comparing BESO with other topology optimization methods

Currently, the SIMP method has demonstrated its effec- tiveness in a broad range of examples and its algorithm receives extensive acceptance due to its computational efficiency and conceptual simplicity (Bendsøe [1989](#_bookmark14); Zhou and Rozvany [1991](#_bookmark38); Sigmund [2001](#_bookmark31); Bendsøe and Sigmund [2003](#_bookmark15)). But the SIMP method using a given penalty exponent *p* may result in a local optimum with grey regions. To avoid

**Fig. 2** Design domain of a long cantilever

local optima, the continuation method must be applied by gradually increasing the penalty exponent (Rozvany et [al.](#_bookmark29) [1994](#_bookmark29)) or gradually decreasing the filter radius ([Sigmund](#_bookmark30) [1997](#_bookmark30)). It should be noted that the filter scheme is a heuris- tic technique for overcoming the checkerboard and mesh- dependency problems in topology optimization. Therefore, it is better to compare BESO with SIMP algorithms at two levels—one without the mesh-independency filter and the other with it.

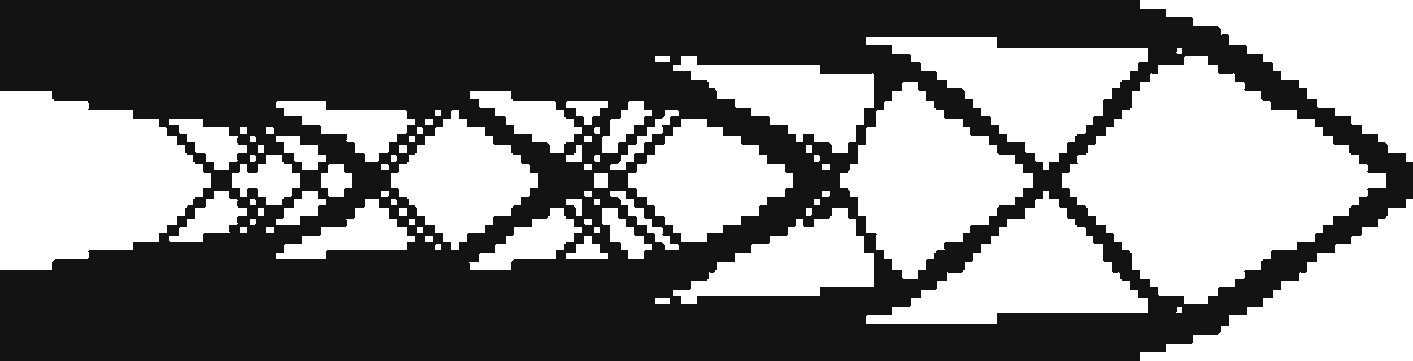
A long cantilever shown in Fig. [2](#_bookmark3) is selected as a test example because it involves a series of bars broken during the evolution process of the ESO/BESO topology. A con- centrated load *F* = 1 N is applied downward in the middle

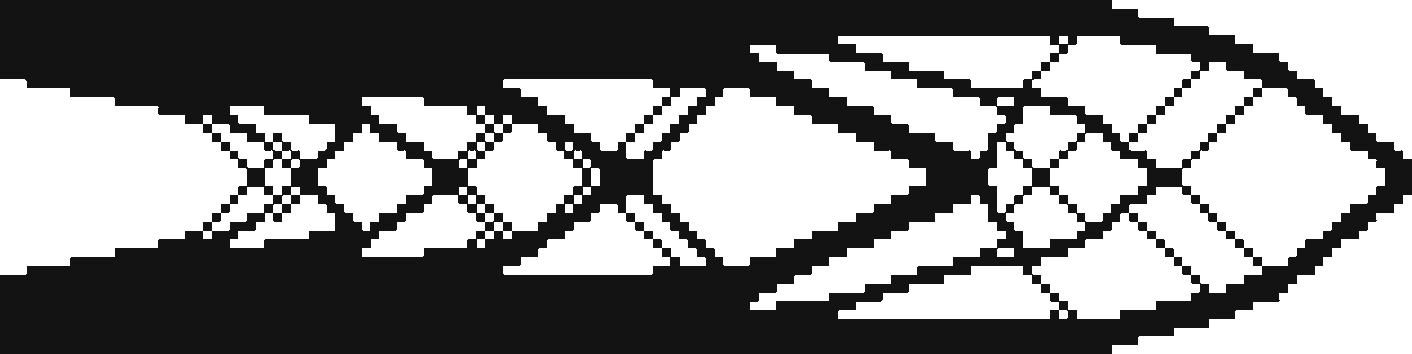
of the free end. Young’s modulus *E* = 1 MPa and Poisson’s

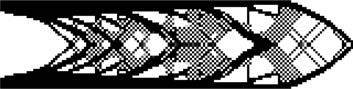
ratio *v* = 0*.*3 are assumed. The design domain is discretized with 160 × 40 four node plane stress elements.

* 1. Comparison of topology optimization algorithms without a mesh-independency filter

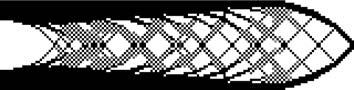
The used parameters and final solutions for various topol- ogy optimization algorithms are listed in Table [1](#_bookmark4). Without using a filter, the topologies obtained from these methods are quite different. It is difficult to tell which topology is the best unless the value of the final objective function is com- pared. It is observed that the continuation method produces the lowest value of *C* among all the optimization methods although it takes the largest number of iterations. ESO and soft-kill BESO require much less iterations and result in mean compliances that are close to that of the continua- tion method. Note that hard-kill BESO without the filter degenerates to ESO. The final mean compliance from the SIMP method with *p* = 3 is higher than the one from





*p* = 3*.*0

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Table 1** Comparison of topology optimization methods without a mesh-independency filter | | | | | |
| Optimization | | Total | Solutions | Errora | |
| parameters | | iteration |  | (%) | |
| ESOb | *ER* = 1% | 67 | *C* = 188*.*91 Nmm | | 4.12 |
| Soft-kill BESO *ER* = 2% 44 *C* = 183*.*25 Nmm 1.00 | | | | | |
|  | *p* = 3*.*0 |  |  | |  |
| SIMP | *move* = 0*.*02 | 37 | *C* = 196*.*48 Nmm | | 8.29 |
| Continuation | *pinitial* = 1 | 337 | *C* = 181*.*44 Nmm | | – |
|  | *6p* = 0*.*1 |  |  |  |  |

*pend* = 5*.*0

aThis refers to the error of the mean compliance C as compared to the result of the continuation method

bHard-kill BESO without a mesh-independency filter degenerates to ESO

other methods because it converges to a local optimum with intermediate elements.

It should be noted that the ESO method usually requires a finer mesh, especially for a problem with a low final volume fraction. Normally, a smaller *ER* results in a better solution. The computational efficiency of ESO highly depends on the selected parameters such as evolutionary ratio *ER* and the mesh size. In most cases, the ESO method using a small *ER* and a fine mesh can provide a good solution. This is a merit of the original ESO procedure.

Compared to ESO, soft-kill BESO and SIMP methods are more stable and less dependent on the used parameters although a relatively fine mesh is still required by the soft- kill BESO method. Provided that the penalty exponent *p* is large enough, both soft-kill BESO and SIMP methods can produce good solutions in most cases since the final solu- tions from these two methods meet the respective optimality criteria.

* 1. Comparison of topology optimization algorithms with a mesh-independency filter

The above problem is reanalyzed using the four topol- ogy optimization methods but this time with a mesh- independency filter. Table [2](#_bookmark5) lists the used parameters and solutions obtained from various topology optimization algo-

rithms. It can be seen that the four topology optimization algorithms produce very similar topologies except that the SIMP design has some grey areas of intermediate mate- rial densities. For practical applications, the topologies in Table [2](#_bookmark5) with a clear definition of each member are far more useful than those shown in Table [1](#_bookmark4).

The mean compliances of both hard-kill and soft-kill BESO solutions are very close to that of the continuation method. However, the continuation method requires more than five times as many iterations as the BESO method. Although hard-kill BESO takes slightly more iterations than soft-kill BESO, the former algorithm is actually the quick- est because the hard-killed elements are not included in the finite element analyses. Again, the SIMP method with *p* 3 converges to a local optimum with a highest mean compliance.

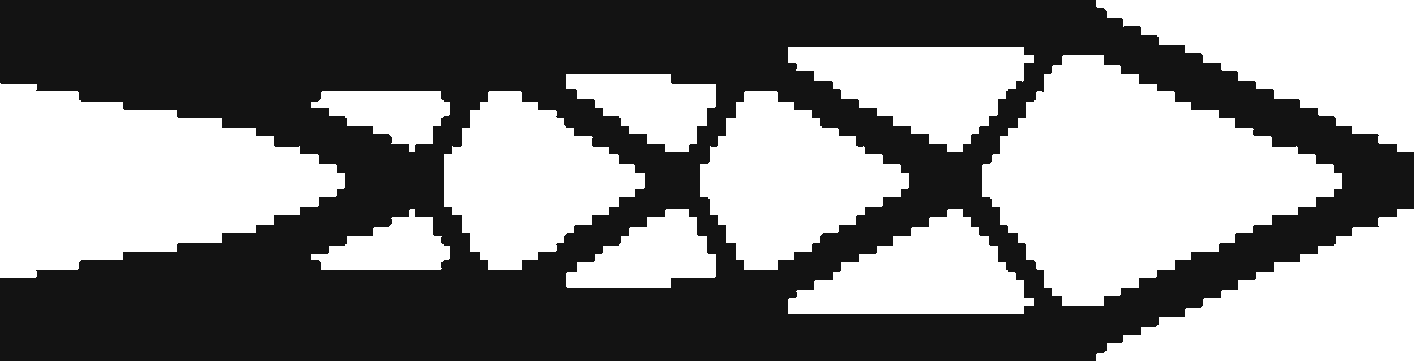
=

Figure [3](#_bookmark6) shows the evolution histories of the objective function using the four topology optimization methods. The mean compliances for both hard-kill and soft-kill BESO methods increases, with occasionally abrupt jumps (due to breaking up of some bars), as the total volume gradually decreases. After about 35 iterations the volume fraction reaches its target of 50%. In subsequent iterations, while the volume remains unchanged the mean compliance gradu- ally converges to a constant value. Different from the BESO methods, both SIMP and continuation methods have the volume constraint satisfied all the time. While the volume

**Table 2** Comparison of topology optimization methods with a mesh-independency filter

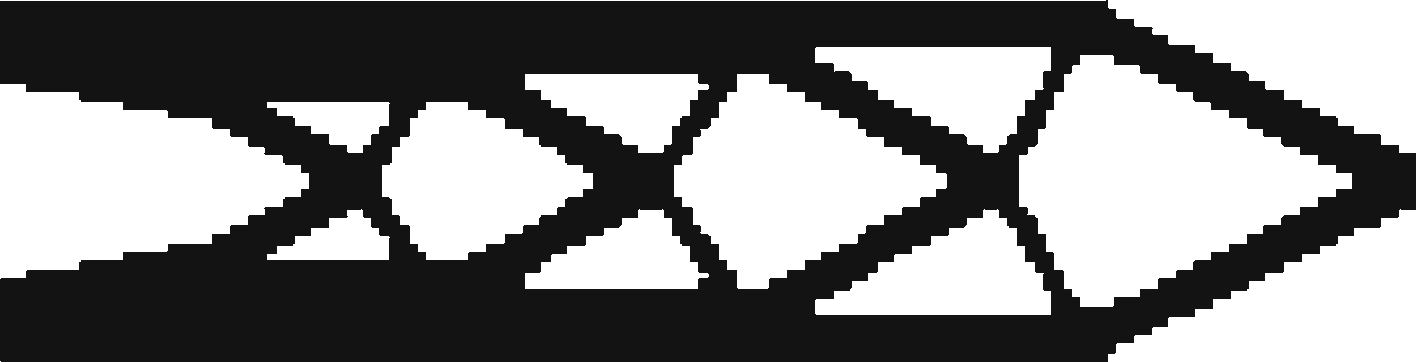
Optimization Total Solutions Errora

parameters iteration (%)

Hard-kill BESOb *ER* = 2% 52 *C* = 181*.*79 Nmm 0.61

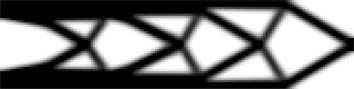
*AR*max = 50%

*r*min = 3*.*0 mm

Soft-kill BESO *ER* = 2% 46 *C* = 181*.*71 Nmm 0.56

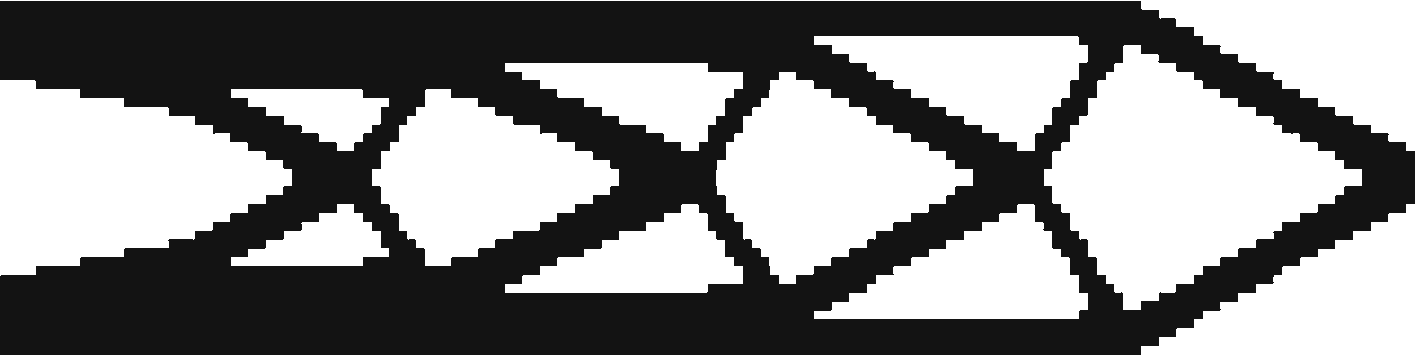
*p* = 3*.*0

*r*min = 3*.*0 mm

SIMP *move* = 0*.*02 44 *C* = 201*.*70 Nmm 11.63

*p* = 3*.*0

*r*min = 3*.*0 mm

Continuation *rini* = 3*.*0 mm 267 *C* = 180*.*69 Nmm –

min

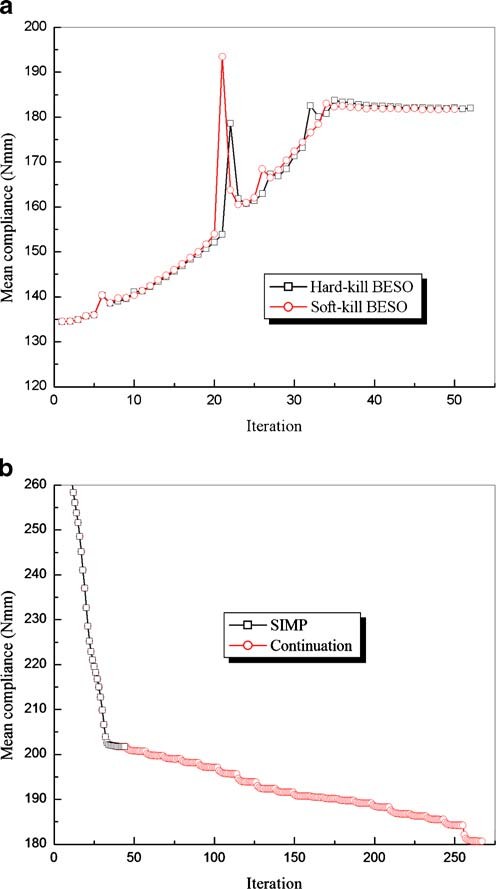
*6r*min = 0*.*1 mm

*rend* = 1*.*0 mm

min

aThis refers to the error of the mean compliance C as compared to the result of the continuation method

bThe penalty tends to infinity for hard-kill BESO method



**Fig. 3** Evolution histories of the mean compliance **a** hard-kill and soft- kill BESO methods; **b** SIMP and continuation methods

is kept constant from the very beginning, the mean compli- ances in SIMP and continuation methods decrease gradually until a convergence criterion is satisfied.

# Discussion on Zhou and Rozvany ([2001](#_bookmark39)) example

* 1. Introduction of Zhou and Rozvany ([2001](#_bookmark39)) example

The structure shown in Fig. [4](#_bookmark8)a is used by Zhou and Rozvan[y](#_bookmark39) ([2001](#_bookmark39)) to show the breakdown of hard-kill optimization methods, such as ESO/BESO. In the example, Young’s modulus is taken as unity and Poisson’s ratio as zero. The mean compliance of the ground structure is about 194. If the design domain is discretized using 100 four node plane

stress elements, the element in the vertical tie will have the lowest strain energy density. Thus, hard-kill ESO/BESO will remove that element from the ground structure and result in the design as shown in Fig. [4](#_bookmark8)b with a mean com- pliance of 2186. This value is much higher than that of any intuitive design obtained by removing one element from the horizontal beam.

After removing an element in the vertical tie, the resul- tant structure becomes a cantilever where the vertical load is transmitted by flexural action. The region with the highest strain energy density is at the left-bottom of the cantilever. According to the BESO algorithm, an element may be added in that region rather than recovering the removed element in the vertical tie.

Therefore, Zhou and Rozvan[y](#_bookmark39) ([2001](#_bookmark39)) conclude that hard- kill optimization methods such as ESO/BESO may produce a highly non-optimal solution. In fact, soft-kill optimization algorithms such as the level set method using continuous design variables may also produce a similar result (Norato et [al.](#_bookmark23) [2007](#_bookmark23)). To overcome this problem, the essence of such a solution needs to be examined first.

* 1. Is it a non-optimal or a local optimal solution?

Obviously, the answer cannot be easily found by simply comparing the values of the objective function. Let us recon- sider the above example with a volume fraction of 96%. Hard-kill optimization methods such as ESO will remove the four elements from the vertical tie as shown in Fig. [4](#_bookmark8)c. This design is certainly far less efficient than an intuitive design which removes four elements from the horizontal beam.

It is known that the SIMP method with continuous design variables guarantees that its solution should be at least a local optimum. Therefore, this topology optimization prob- lem is tested by the SIMP method starting from an initial

guess design (with *xi* = 1 for all elements in the horizontal

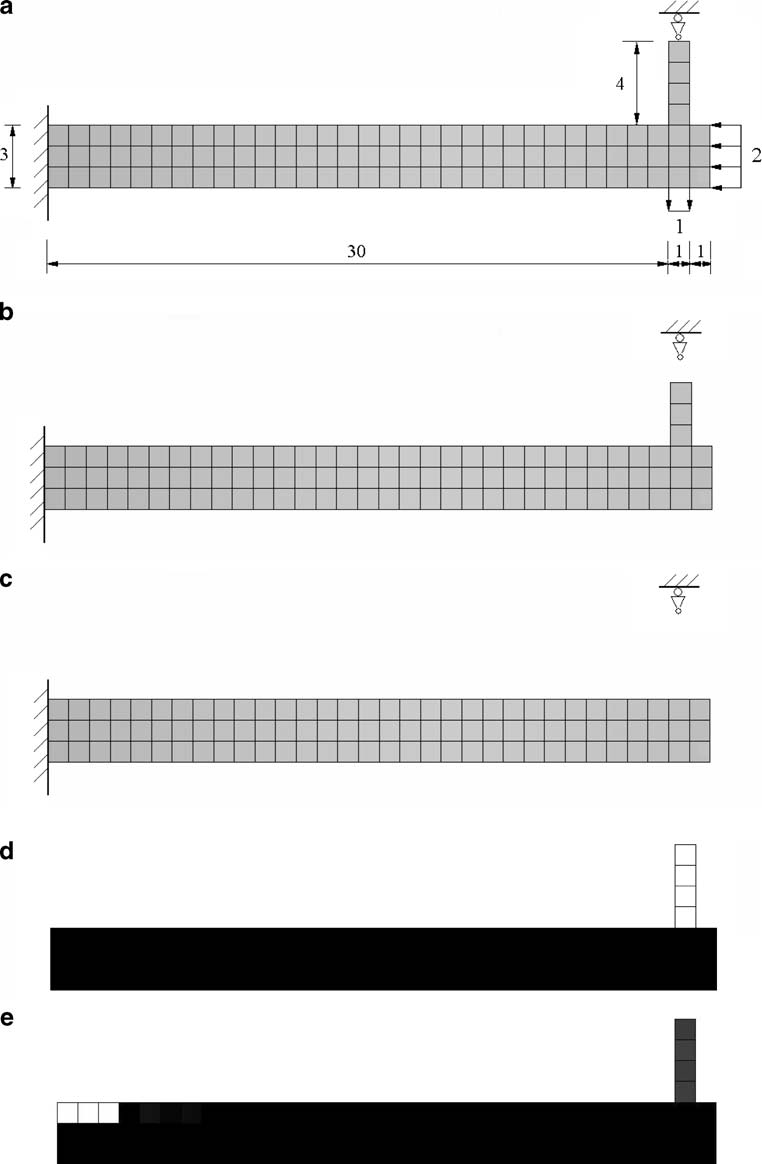
= =

beam and *xi x*min 0*.*001 for the four elements in the

vertical tie). It is found that when *p* 3*.*1 the final solu- tion converges to the structure shown in Fig. [4](#_bookmark8)d, which is exactly the same as the initial guess design. Because *x*min is small, the SIMP solution in Fig. [4](#_bookmark8)d can be considered to be identical to the ESO/BESO solution in Fig. [4](#_bookmark8)c. These results demonstrate that the above solutions from ESO/BESO and SIMP are essentially a local optimum rather than a non- optimum. Theoretically it may be more appropriate to call such a solution *a highly inef f icient local optimum* than a non-optimum.

≥

The occurrence of the above 0/1 local optimal design is caused by the large penalty *p* in the optimization algorithms. Hard-kill ESO/BESO methods have an equivalent penalty of infinity and therefore fail to obtain a better solution once they reach the highly inefficient local optimum. Similarly,

**Fig. 4** The example in Zhou and Rozvan[y](#_bookmark39) ([2001](#_bookmark39)) **a** boundary and loading conditions;

**b** ESO/BESO design for

*V f* 99%. **c** a highly inefficient local optimum for *V f* 96% from ESO/BESO; **d** a highly inefficient local optimum for *V f* 96% from

=

=

=

SIMP when *p* 3*.*1; **e** optimal solution for *V f* 96% from the continuation method

=

≥

=

the soft-kill BESO method with a finite penalty may also fail because a large penalty ( *p* 1*.*5) is normally required for topology optimization.

≥

=

The exact value of the penalty *p* that is large enough to cause a local optimum is dependent upon the optimiza- tion problem. For the original Zhou and Rozvan[y](#_bookmark39) ([2001](#_bookmark39))

example given in Fig. [4](#_bookmark8)a, the SIMP method will produce a much more efficient solution than the one shown in Fig. [4](#_bookmark8)d when *p* 3 is used. However, if we modify the original problem slightly by reducing the vertical load from 1 to 0.5, the SIMP method with *p* 3 will again result in the highly inefficient local optimum shown in Fig. [4](#_bookmark8)d.

* 1. Avoidance of a local optimum within optimization algorithms

It is well-known that most topology optimization problems are not convex and may have many different local optima. At the same time, most global optimization methods seem to be unable to handle problems of the size of a typical topol- ogy optimization problem (Bendsøe and S[igmund](#_bookmark15) [2003](#_bookmark15)). As shown in the above section, the ESO/BESO method and the SIMP method fail to ensure a global optimum and the resulting topologies depend on choices of optimization parameters and initial guesses.

Based on the experience, the local optimum can be avoided using the continuation method by gradually increas- ing the penalty exponent from 1 (Rozvany et [al.](#_bookmark29) [1994](#_bookmark29)). For this particular problem, the continuation method with *6p* 0*.*1 produces an optimal solution shown in Fig. [4](#_bookmark8)e after about 700 iterations. The continuation method fails to produce a pure 0/1 global solution due to the numeric overflow although it successfully avoids the above highly inefficient local optimum. Theoretically, a global optimum cannot be guaranteed even for the continuation method as noted by Stolpe and Svanber[g](#_bookmark33) ([2001](#_bookmark33)).

=

Therefore, it is unfair to expect the ESO/BESO methods to overcome a local optimum while other well-established methods would fail as well. To completely solve this prob-

studies demonstrated that an optimal design can be obtained with a fine mesh.

It should be noted that the mesh refinement causes the change of the original optimization problem of finding a global optimum under a given mesh as argued by Rozvan[y](#_bookmark27) ([2009](#_bookmark27)). But, checking the boundary condition outside the ESO/BESO algorithms is a conservative but effective way to detect the occurrence of a highly inefficient 0/1 local optimum for this particular problem.

# Extension of BESO to displacement constraint problem

* 1. The optimization problem and sensitivity number

The main characteristic of the ESO/BESO procedure is to gradually change the structural volume. Thus a volume con- straint rather than a displacement constraint can be easily implemented. As observed by Rozvan[y](#_bookmark27) ([2009](#_bookmark27)), the previous BESO procedure cannot be readily used for optimization problems with constraints other than the structural volume. The topology optimization problem with a displacement constraint may be stated as

Minimize: *f (x)* = *V* = . *Vi xi*

*N*

lem, further research is required for all topology optimiza- tion methods, not just the ESO/BESO methods.

Subject to: *uj* = *u*∗*j x**i* = *x*min or 1

*i* =1

(9)

4.4 Avoidance of a local optimum outside ESO/BESO optimization algorithms

Nonetheless, it is necessary to find a solution outside the ESO/BESO algorithms to avoid this type of highly ineffi- cient 0/1 local optima. Fortunately, a 0/1 highly inefficient local optimum can be easily identified even by inspection. In the above example, the cantilever is a substructure of the ground structure and its optimal solution may be a 0/1 local optimum solution of the whole structure. Therefore, 0/1 local optima widely exist in topology optimization problems for a statically indeterminate structure.

Huang and [Xie](#_bookmark20) ([2008](#_bookmark20)) proposed that this inefficient local optimum can be detected by checking the boundary con- ditions for a statically indeterminate structure after each

*p*−1

where *u j* and *u*∗*j* denote the *j* th displacement and its constraint respectively.

In order to solve this problem using the BESO method, the displacement constraint should be added to the objective function by introducing a Lagrangan multiplier *λ* as

*N*

. . Σ

*f*1*(x)* = *Vi xi* + *λ uj* − *u*∗*j* (10)

*i* =1

It can be seen that the modified objective function would be equivalent to the original one and the Lagrangian multiplier can be any constant if the displacement constraint is sat- isfied. With the SIMP model, the derivative of the modified objective function is

iteration. If a breakdown of boundary support is detected

*∂ f*1

= *V*

*∂uj*

+ *λ* = *V*

− *λpx* **u***T* **K**0**u**

(11)

before a satisfactory solution is obtained, it may well indi- cate that thereafter the solution may be (but not always) a

*∂xi*

*i ∂xi i*

*i ij i i*

highly inefficient local optimum and the current optimiza- tion process should be stopped immediately.

Then, the problem is re-calculated with a fine mesh to avoid breakdown of the boundary. Edwards et [al.](#_bookmark18) ([2007](#_bookmark18)) and Huang and X[ie](#_bookmark20) ([2008](#_bookmark20)) in their parallel but independent

where the sensitivity of the *j* th displacement is calculated using the adjoint method (Bendsøe and S[igmund](#_bookmark15) [2003](#_bookmark15)). **u***ij* is the virtual displacement vector of the *i* th element resulted from a dummy load whose *j* th component is equal to unity and all others are equal to zero. When a uniform

mesh is used, the relative ranking of sensitivity of each ele- ment can be defined by the following sensitivity number

the topology, the following equation is adopted to determine the structural volume for the next iteration.

*αi* =− . − *Vi* Σ = *x* **u***T* **K***i* **u***i* (12)

1 *∂ f*1 *p*−1

min

*V k (*1 + *E R) , V c*

when *V k* ≤ *V c*

.max .*V k (*1 − *E R) , V c*Σ when *V k > V c*

*λp ∂xi*

*i ij*

*V k*+1 = . Σ

(16)

Therefore, the sensitivity numbers for solid and void ele- ments are expressed explicitly by

⎧⎨**u***T* **K**0**u***i xi* = 1

*ij*

*i*

*i* = *p*−1

The above equation ensures that the volume change in each iteration should be less than the prescribed the evolution-

*α*

⎩*x* **u***T* **K**0**u**

min

*ij*

*i*

*i*

*i*

*x* = *x*

(13)

ary rate, *ER*, which defines the maximum variation of the

* 1. Determination of structural volume

min

structural volume in a single iteration.

For the present optimization problem, the structural volume is to be minimized and determined according to the pre- scribed displacement constraint. The sensitivity analysis of the constraint displacement is given as

* 1. Example

To demonstrate the proposed method, a topology optimiza- tion problem for a beam which is supported at both ends and vertically loaded ( *P* = 100 N) in the middle of its

*p*−1

*∂uj*

= *px* **u***T* **K**0**u**

(14)

lower edge as depicted in Fig. [5](#_bookmark11)a is considered. Due to

*∂xi*

*i ij i i*

the symmetry, the computation is performed for the right half of the domain with 120 × 40 four node plane stress

Thus, the *j* th displacement in the next iteration, *uk*+1, can

elements. The linear material is assumed with Young’s mod-

be estimated by the *j* th displacement in the current

*uk* , as

*j*

*j*

iteration,

ulus *E* 100 GPa and Poisson’s ratio *v* 0*.*3. It is required that the maximum deflection of the beam should not exceed

0.2 mm under the given load. The BESO parameters are

= =

*k*+1

*k* . *du*

*ER* = 2%, *x*min = 0*.*001, *p* = 3 and *r*min = 1*.*5 mm.

From the above equation, the threshold of the sensitivity number as well as the corresponding volume, *V c*, can be easily determined by letting *uk*+1 *u*∗. However, the resul- tant volume *V c* may be much larger or far smaller than that of the current design. In order to have a gradual evolution of

*k*

*u j* ≈ *u j* +

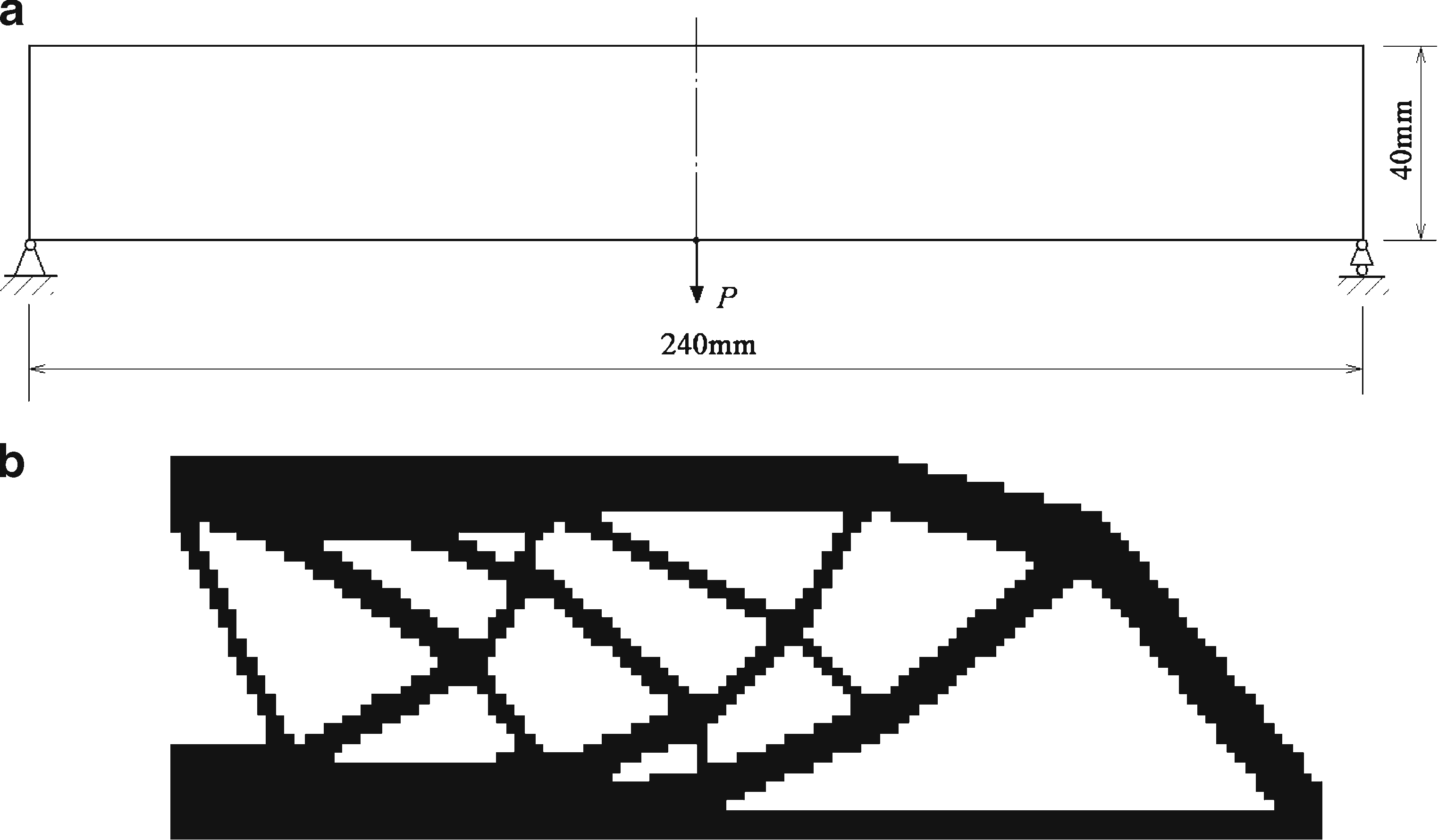
*dxi 6xi* (15)

The final design shown in Fig. [5](#_bookmark11)b has 45% volume of the initial full design. The maximum deflection is 0.1997 mm

=

which is very close to the prescribed constraint limit,

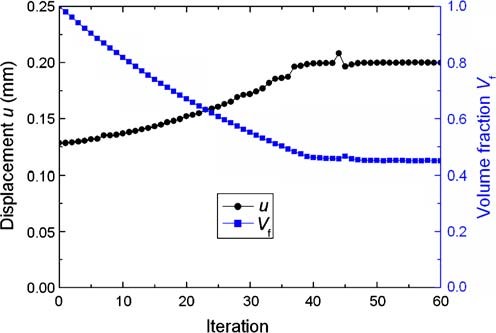
0.2 mm. Evolution histories of the volume fraction and the constraint displacement are shown in Fig. [6](#_bookmark12). It can be seen that both volume fraction and the constraint displacement stably converge after 45 iterations.

**Fig. 5** The optimization problem for minimizing the structural volume against the maximum deflection

*u*\* 0.2 mm **a** design domain of a beam; **b** optimal design using “soft-kill” BESO

=

with *p* = 3*.*0



**Fig. 6** Evolutionary histories of the volume fraction and maximum deflection

In the above optimization problem, the displacement constraint is applied to the location of the external force. Therefore the optimal solution is equivalent to the solution of the minimum compliance problem with a corresponding volume constraint. We have optimized the above beam by minimizing the mean compliance subject to a volume con- straint, *V f* 0.45. The mean compliance of the resulted design is 10.07 N mm which is very close to that of the pre- vious solution with a displacement constraint, 9.985 N mm. It indicates that *V f* 0.45 is the true minimum value when the displacement constraint is set to be 0.2 mm.

=

=

5.4 Further comments

It should be noted that the above procedure cannot be directly used for a hard-kill BESO. In the hard-killed BESO method, the penalty exponent *p* is infinite and therefore the derivative of displacement in ([14](#_bookmark10)) becomes infinite too for the solid element. An easy way to circumvent this problem is to use the following algorithm to determine the struc- tural volume by comparing the current displacement with its constraint value as

multiplier. However, the Lagrangian multiplier must be cal- culated when multiple constraints are present in the opti- mization problem as discussed by Huang and X[ie](#_bookmark22) ([2009b](#_bookmark22)). In such a case, it is almost impossible to use a hard-kill BESO method.

# Conclusion

This paper has presented the most recent developments in the BESO method for stiffness optimisation problems. It shows that the current BESO method stably converge to a optimal solution with high computational efficiency. The optimal solutions of soft-kill and hard-kill BESO methods with a mesh-independency filter agree well with those of the SIMP and continuation methods for stiffness optimiza- tion problems. This paper also demonstrates that the current BESO method can be easily extended to other constraint such as a displacement constraint.

Through numerical experiments, it is shown that the Zhou and Rozvan[y](#_bookmark39) ([2001](#_bookmark39)) “non-optimal” solution resulting from the hard-kill optimisation methods like ESO/BESO is actually a local optimum which may also occurs in many soft-kill optimization methods. The occurrence of the local optimum is attributed to the artificial material penalty rather than hard-killing elements. To totally dismiss the merit of ESO/BESO methods for topology optimization of con- tinuum structures by simply citing the Zhou and Rozvan[y](#_bookmark39) ([2001](#_bookmark39)) example, as some commentators have chosen to do, is hardly justified.

The main aim of this paper is to demonstrate the effec- tiveness of the current BESO and to answer the critical com- ments on the original ESO type methods raised by Rozvany ([2009](#_bookmark27)). Even so, we agree with Rozvan[y](#_bookmark27) ([2009](#_bookmark27)) that we should cautiously remove elements (design variables) from the design domain. In other words, for a new topology opti- mization problem, it would be prudent to develop a soft- kill BESO method first and then explore the possibility of hard-killing elements.

⎪⎧ .

. *ui* −*u*∗ ΣΣ

**Acknowledgments** The authors wish to acknowledge the finan-

⎪⎪⎪⎪⎪max

⎪⎪⎪

*V i (*1 − *E R) , V i*

1− *j*

*j*

*u*∗*j*

*j*

.

ΣΣ

(17)

cial support from the Australian Research Council for carrying out this work.

when *ui*

*V i* +1 =

.

*j*

⎨

*> u*∗

⎪⎪⎪min *V (*1 + *E R) , V* 1+ *u*∗

⎪⎪⎪

*i*

*i*

*u*∗*j* −*ui*

*j*

**Appendix**

⎪⎪⎪⎩ *i* ∗ *j*

when *u j* ≤ *u j*

Numerical tests indicate that the above algorithm works well for the hard-kill BESO method.

In the present optimization problem with a displacement constraint, it is unnecessary to calculate the Lagrangian

This appendix contains a soft-kill BESO Matlab code that can be used to solve simple 2D stiffness optimization prob- lems. The code is developed based on the 99 line SIMP code [(Sigmund](#_bookmark31) [2001](#_bookmark31)). The design domain is assumed to be rectangular and discretized using four node plane stress elements. Here, a cantilever is taken as an example. Other

structures with different loading and boundary conditions can be solved by modifying lines 80–84 of the code. The input data are:

*nelx* total number of elements in the horizontal direction.

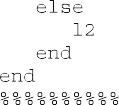
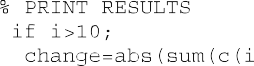
*nely* total number of elements in the vertical direction. *volfrac* volume fraction which defines the ratio of the final volume and the design domain volume.

*er* evolutionary rate, normally 0.02.

*rmin* filter radius, normally 3 (or the size of several elements).



















# References



Bendsøe MP (1989) Optimal shape design as a material distribution problem. Struct Optim 1:193–202



Bendsøe MP, Sigmund O (2003) Topology optimization: theory, methods and applications. Springer, Heidelberg

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

Cui C, Ohmori H, Sasaki M (2005) Structural design by extended ESO method. In: Proceedings of frontiers of computational sciences symposium, Nagoya, Japan, 11–13 October 2005, pp 149–156

Edwards CS, Kim HA, Budd CJ (2007) An evaluative study on ESO and SIMP for optimizing a cantilever tie-beam. Struct Multidisc Optim 34(5):403–414

Huang X, Xie YM (2007) Convergent and mesh-independent solutions for the bi-directional evolutionary structural optimization method. Finite Elem Anal Des 43:1039–1049

Huang X, Xie YM (2008) A new look at the ESO/BESO optimization methods. Struct Multidisc Optim 35(1):89–92

Huang X, Xie YM (2009a) Bi-directional evolutionary topology opti- mization of continuum structures with one or multiple materials. Comput Mech 43(3):393–401. doi:[10.1007/s00466-008-0312-0](http://dx.doi.org/10.1007/s00466-008-0312-0)

Huang X, Xie YM (2009b) Evolutionary topology optimization of continuum structures with a local displacement constraint. Struct Multidisc Optim 40:409–416. doi:[10.1007/s00158-009-0382-4](http://dx.doi.org/10.1007/s00158-009-0382-4)

Norato JA, Bendsøe MP, Harber RB, Tortorelli DA (2007) A topologi- cal derivative method for topology optimization. Struct Multidisc Optim 33:375–386

Ohmori H, Futai H, Iijima T, Muto A, Hasegawa Y (2005) Application of computational morphogenesis to structural design. In: Proceed- ings of frontiers of computational sciences symposium, Nagoya, Japan, 11–13 October 2005, pp 45–52

Querin OM, Steven GP, Xie YM (1998) Evolutionary structural opti- mization (ESO) using a bi-directional algorithm. Eng Comput 15:1031–1048

Querin OM, Young V, Steven GP, Xie YM (2000) Computational efficiency and validation of bi-directional evolutionary struc- tural optimization. Comput Methods Appl Mech Eng 189: 559–573

Rozvany GIN (2009) A critical review of established methods of struc- tural topology optimization. Struct Multidisc Optim 37:217–237

Rozvany GIN, Querin OM (2002) Combining ESO with rigorous optimality criteria. Int J Veh Des 28:294–299

Rozvany GIN, Zhou M, Sigmund O (1994) Optimization of topology. In: Adeli H (ed) Advances in design optimization. Chapman & Hall, London

Sigmund O (1997) On the design of compliant mechanisms using topology optimization. Mech Struct Mach 25:495–592

Sigmund O (2001) A 99 line topology optimization code written in Matlab. Struct Multidisc Optim 21:120–127

Sigmund O, Petersson J (1998) Numerical instability in topology opti- mization: a survey on procedures dealing with checkerboards, mesh-dependencies and local minima. Struct Optim 16:68–75

Stolpe M, Svanberg K (2001) On the trajectories of penalization methods for topology optimization. Struct Multidisc Optim 21: 128–139

Xie YM, Steven GP (1992) Shape and layout optimization via an evolutionary procedure. In: Proceedings of the international

conference computational engineering science. Hong Kong, p 421

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, London

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: topo- logical, 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