Transactions on the Built Environment vol 28, © 1997 WIT Press, www.witpress.com, ISSN 1743-3509

Shape optimal design using GA and BEM

Eisuke Kita & Hisashi Tanie
Department of Mechano-Informatics and Systems,
Nagoya University, Nagoya 464-01, Japan

Abstract

This paper describes a shape optimization scheme of two-dimensional continuum structures by genetic algorithm (GA) and boundary element method (BEM). The profiles of the objects under consideration are represented by Free-Form Deformation (FFD) method. The chromosomes for the profiles are defined by the FFD control points. The population constructed by many chromosomes is modified by the genetic operations such as the selection, the crossover and the mutation in order to determine the profile satisfying the design objective. The boundary element method is employed for estimating the fitness functions. The present method is applied to the shape optimization of a cantilever beam.

1 Introduction

The GA-based schemes for the shape optimization of the continuum structures can be classified into the cell representation scheme[1, 2] and the free-curves representation scheme[3, 4]. In the cell representation scheme, the object domain is divided into small square cells and then, the binary parameter is specified to each cell. The chromosome for the profile of the object is defined by the series of the binary parameters. The chromosomes are very long when many cells are employed for the shape representation. Besides, in the actual design and manufacturing processes, the free-curves representation of the objects (e.g., machine and structures) is necessary and therefore, the data of the zigzag-shaped final profiles must be translated to the data of the free curves representation. These difficulties can be overcome partially by using the free curve representation scheme. The profile of the object domain is represented by the free curves and then, the chromosome is defined by the coordinates of the control points of the curves. In this case, the entire profile can be represented by relatively small number of control points. Moreover, the obtained profile can be used directly in the design and the manufacture process. However, in the ordinary schemes, finite element method is employed for estimating the fitness function. The finite element mesh is distorted by the successive shape modification and therefore, the computational efficiency may become worse. In order to overcome the above-mentioned difficulties, this paper presents the GA-based optimization scheme using the boundary element method. The profiles of the objects are represented by the Free Form Deformation method (FFD). Since the FFD is often employed in the computer graphics, the obtained results can be employed directly in the design and the manufacturing process. The fitness functions are estimated by the BEM and therefore, the distortion of the mesh by the shape modification is not so terrible.

In this paper, firstly, the profile representation by the FFD is explained and the chromosome is defined. Then, the algorithm of the present method is explained. Finally, the present method is applied to the shape optimization of the cantilever beam.

2 Free Form Deformation (FFD)

In the Free Form Deformation method (FFD)[5], the local coordinate system $s-t-u$ on the parallelepiped region is imposed firstly. The position vector of an arbitrary point $\boldsymbol{X}$ is represented as:

$$\boldsymbol{X} = \boldsymbol{X}_0 + s\boldsymbol{S} + t\boldsymbol{T} + u\boldsymbol{U} \tag{1}$$

where $\boldsymbol{X}_0$ denotes the position vector of the origin and then, $\boldsymbol{S}, \boldsymbol{T}$ and $\boldsymbol{U}$ the unit vectors in the S, T and U directions, respectively. s, t and u, which indicate the coordinates of the position $\boldsymbol{X}$ in the local coordinate systems, are calculated as follows:

$$s = \frac{\boldsymbol{T} \times \boldsymbol{U} \cdot (\boldsymbol{X} - \boldsymbol{X}_0)}{\boldsymbol{T} \times \boldsymbol{U} \cdot \boldsymbol{S}} \tag{2}$$

$$t = \frac{\boldsymbol{S} \times \boldsymbol{U} \cdot (\boldsymbol{X} - \boldsymbol{X}_0)}{\boldsymbol{S} \times \boldsymbol{U} \cdot \boldsymbol{T}} \tag{3}$$

$$u = \frac{\boldsymbol{S} \times \boldsymbol{T} \cdot (\boldsymbol{X} - \boldsymbol{X}_0)}{\boldsymbol{S} \times \boldsymbol{T} \cdot \boldsymbol{U}} \tag{4}$$

Next, a grid of control points on the parallelepiped is imposed. When the objects are not deformed, the control points lie on the lattice position. The position vector of the arbitrary control point $\boldsymbol{P}_{ijk}$ is given as:

$$\boldsymbol{P}_{ijk} = \boldsymbol{X}_0 + \frac{i}{l}\boldsymbol{S} + \frac{j}{m}\boldsymbol{T} + \frac{k}{n}\boldsymbol{U} \tag{5}$$

where i, j and k are integer numbers taken as $0 \leq i \leq l, 0 \leq j \leq m$ and $0 \leq k \leq n$, respectively. These form $l+1$ planes in the S direction, $m+1$ planes in the T direction and $m+1$ planes U direction, respectively.

The profile of the object is deformed by moving the control points from the lattice position. Assuming that $\boldsymbol{X}$ will move to $\boldsymbol{X}_{FFD}$, $\boldsymbol{X}_{FFD}$ is given as:

$$\boldsymbol{X}_{FFD} = \sum_{i=0}^{l}\sum_{j=0}^{m}\sum_{k=0}^{n} B_{i,l}(s)B_{j,m}(t)B_{k,n}(u)\boldsymbol{P}_{ijk} \tag{6}$$

where

$$B_{i,n}(t) = {}_nC_i(1-t)^{n-i}t^i \tag{7}$$

${}_nC_i$ is the combination:

$${}_nC_i = \frac{n!}{i!(n-i)!} \tag{8}$$

3 Algorithm

We describe here the algorithm of the present method. The detailed explanation of the genetic algorithm is shown in some references, e.g., [6, 7].

3.1 Optimization problem

The design objective is to minimize the weight of the object under consideration by changing the profile of the object. As the constraint conditions, we consider that the maximum stress $\sigma_{\max}$ on the boundary is smaller than the allowable stress of the material σ_c and that the profile of the object does not cross. The objective function f and the constraint conditions h_1 and h_2 are given as:

$$\min f = \frac{A}{A_0} \tag{9}$$

subject to

$$h_1 = \sigma_{\max} - \sigma_c \leq 0 \tag{10}$$

$$h_2 = g_c = 0 \tag{11}$$

where A_0 denotes the area of the initial profile. g_c is the function related to the crossing of the profile; $g_c = 1$ if the profile cross and $g_c = 0$ if not so.

Besides, the position vectors of the FFD control points $\boldsymbol{P}_{ijk}$ are taken as the design variables. The constraint conditions for the design variables are specified in order to restrict the design space.

3.2 Genetic coding

(1) Chromosome The position vectors of the control points $\boldsymbol{P}_{ijk}$ are considered as the genes and then, the chromosomes for the profile of the object under consideration are defined as:

$$\boldsymbol{P}_{000}\boldsymbol{P}_{001}\boldsymbol{P}_{010}\boldsymbol{P}_{100}\cdots\boldsymbol{P}_{lmn} \tag{12}$$

The length of the chromosomes is $(l+1)\times(m+1)\times(n+1)$, which is invariant during the process.

(2) Fitness function The fitness function is defined by introducing the penalty function α;

$$fitness = 1 - \left\{\frac{A}{A_0} + \alpha[H(\sigma_{\max}, \sigma_c) + H(g_c, 0)]\right\} \tag{13}$$

where H is the step function and α is taken as 10000.

3.3 Genetic operations

(1) Selection The ranking selection operation is employed in this study. The ranking selection operation selects the parents according to the ranks of the function values. The ranks of the individuals are specified according to the magnitudes of their fitness values. While the highest selection rate is specified to the individual of the highest rank, the lowest rate is to the one of the lowest rank that has, at least, one offspring. The rates of the other individuals are determined by linear interpolation of them. In addition to the ranking procedure, the elitist model is employed in order to keep the best individual at the next generation.

(2) Crossover The crossover operation swaps some genes of the selected parents in order to create the offspring. The one-point, two-point and uniform crossovers are compared in the numerical examples.

(3) Mutation Normal mutation operation moves randomly the position vectors of the control points. If the magnitude of the movement is large, the mutation destroy the profiles of the objects. Therefore, the maximum value of the movement is specified by the user in advance.

While the normal mutation moves randomly the control points, the directed mutation moves them to the function space where there may exist the better solution. If the mutation improves the fitness function, the directions of the movements of the genes are memorized. Then, at the next generation, the genes are moved slightly in the memorized directions and then, randomly by the usual mutation.

3.4 Flow of present scheme

The algorithm of the basic scheme is as follows:

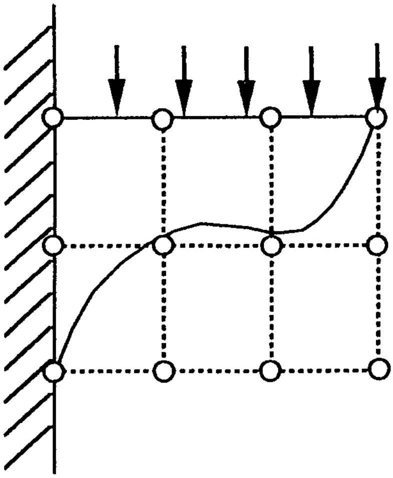

Figure 1: Object under consideration

1. Individuals are generated randomly to construct initial population.
2. Fitness functions of the individuals are estimated by using the BEM.
3. Convergence criterion is estimated. If the criterion is satisfied, the process is terminated. If not so, the process goes to the following steps.
   (a) Crossover operation is performed.
   (b) Mutation operation is performed.
   (c) Population is renewed.
   (d) Go to Step 2.

4 Numerical Example

4.1 Results by scheme 1

A cantilever beam under uniformly distributed load is considered as a numerical example (Fig.1). 12 control points are employed for controlling the profiles. Left and upper segments of the profiles are fixed and therefore, the shape modification is performed by moving 6 control points.

In the scheme 1, the ranking selection, the one-point crossover and the normal mutation operations are adopted. Besides, each population is constructed by 50 individuals. The crossover rate and the mutation rate are

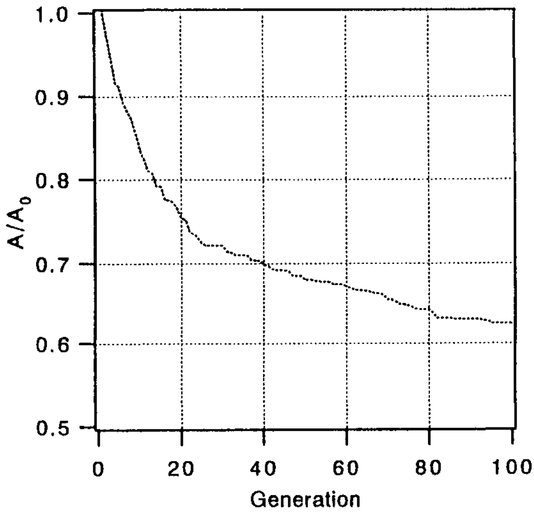

Figure 2: Result by scheme 1

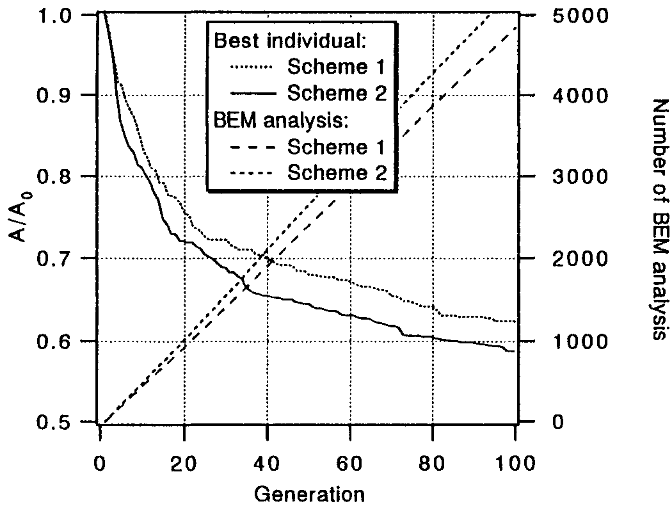

Figure 3: Comparison of schemes 1 and 2

specified as 1 and 0.02, respectively. These rates are invariant during the computation.

Figure 2 indicates the object function values of the best individual at each generation. The performance of the best individual is improved during the process.

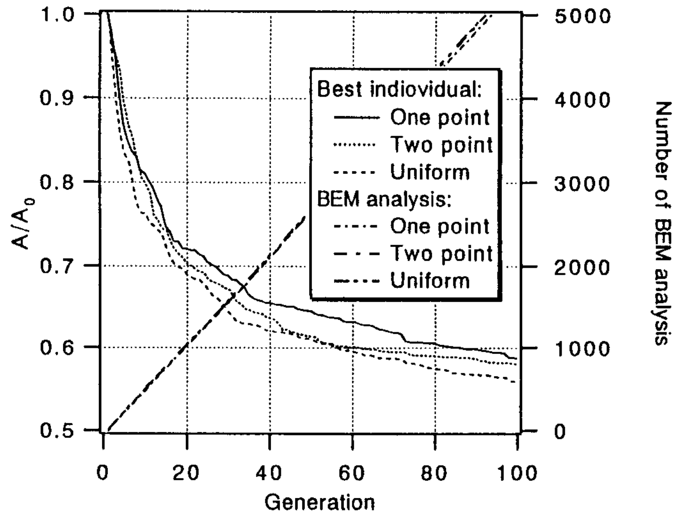

Figure 4: Comparison of crossover operations

4.2 Improvement of computational efficiency

(1) Scheme 2 In the scheme 1, the constraint conditions are not confirmed for each individual when the next generation is constructed. Therefore, the new population includes the individuals which do not satisfy the constraint conditions. In the scheme 2, the new population is constructed by the individuals satisfying the constraint conditions alone. When a offspring individual is generated, the constraint conditions are confirmed. If a offspring individual dose not satisfy the conditions, it is eliminated. Since, in this case, all individuals satisfy the conditions, the following fitness function is adopted.

$$fitness = 1 - \frac{A}{A_0} \tag{14}$$

Figure 3 indicates the object function values of the best individuals at each generation. The scheme 2 produces the better individual than the scheme 1. In the scheme 2, the number of the BEM analyses is larger than the scheme 1. This is because the BEM analyses are repeated until the population is filled with the individuals satisfying the conditions alone.

(2) Improvement of crossover The schemes 1 and 2 adopt one-point crossover operation. The one-point, two-point and uniform crossover operations are compared in order to improve the scheme 2. Figure 4 indicates the object function values of the best individuals at each generation. We notice that the uniform crossover can search the better individual than the other crossover operations.

Transactions on the Built Environment vol 28, © 1997 WIT Press, www.witpress.com, ISSN 1743-3509

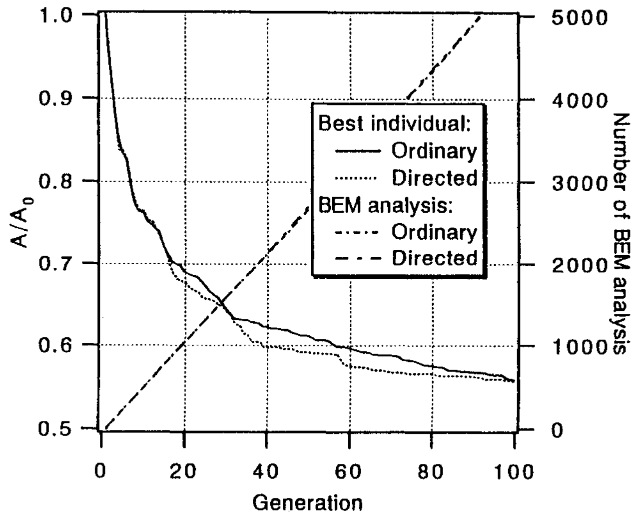

Figure 5: Comparison of mutation operations

The scheme 2 using, instead of the one-point crossover, the uniform crossover is referred to as the scheme 3.

(3) Employment of directed mutation The normal and the directed mutation operations are compared in order to improve the efficiency of the scheme 3. Figure 5 indicates the objective function values of the best individuals at each generation. The scheme using the directed mutation can search the better individual than the scheme using the normal mutation. The scheme 4 is referred to as the scheme 3 using, instead of the normal mutation, the directed mutation. Figure 6 indicates the profile of the best individual at each generation by the scheme 4.

5 Conclusion

This paper presented the GA-based schemes for the shape optimization of the continuum structures. The profile of the object under consideration is represented by the FFD. The position vectors of the FFD control points are considered as the genes and then, the chromosome for the profile is defined by the series of the genes. Therefore, the entire profile of the object can be represented by relatively small number of the genes. Besides, the fitness functions of the individuals are estimated by the BEM. Since, in this case, the distortion of the mesh by the successive shape modification is not so terrible, the accurate estimation of the fitness function can be done.

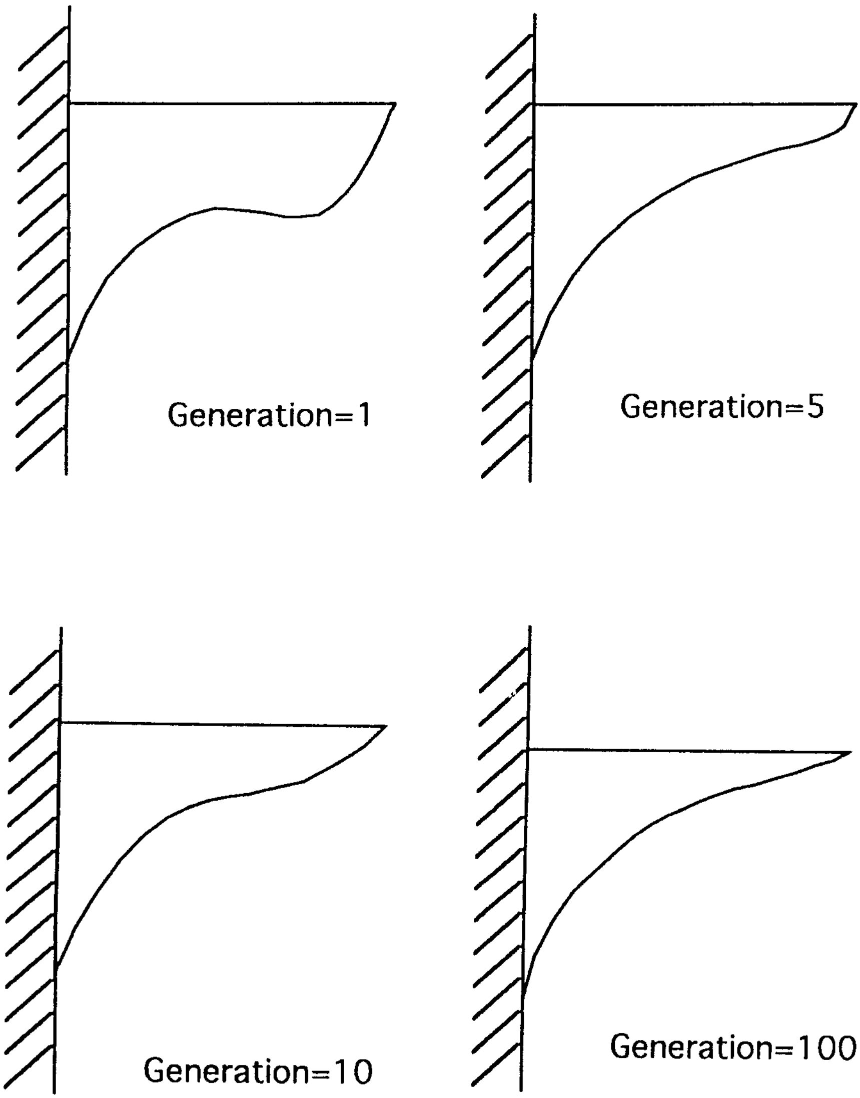

Figure 6: Profiles of best individuals

The present scheme was applied to the shape optimization of the cantilever beam. The results were very satisfactory. However, the computational cost is relatively high. Therefore, some schemes were presented for improving the computational cost.

References

[1] E. Sandgren, E. Jensen, and J. Welton. Topological design of structural components using genetic optimization methods. *Sensitivity Analysis and Optimization with Numerical Methods*, Vol. AMD-Vol.115, pp.31–43, 1990.

[2] E. Sandgren and E. Jensen. Automotive structural design employing a genetic optimization algorithm. *SAE Technical Paper No.920772*, 1992.

[3] H. Watabe and N. Okino. A study on genetic shape design. *Proc. 5th Int. Conf. Genetic Algorithm*, pp. 445–450. Morgan Kaufmann Pub., 1993.

[4] K. Takahashi, Y. Takahashi, and K. Watanabe. 2d shape optimization by genetic algorithms. *Proceeding of 50th Conference of Information Processing Society of Japan*, Vol. 2, pp. 259–260, 1995. (in Japanese).

[5] T. W. Sederberg and S. R. Parry. Free-form deformation of solid geometric models. *Computer Graphics* , Vol. 20, pp. 151–160. ACM SIGGRAPH, 1986.

[6] D. E. Goldberg. *Genetic Algorithms in Search, Optimization and Machine Learning*. Addison Wesley, 1 edition, 1989.

[7] L. Davis. *Handbook of Genetic Algorithms*. Van Nostrand Reinhold, 1 edition, 1991.