

A Constraint Handling Strategy for Bit-Array Representation GA in Structural Topology Optimization

Shengyin Wang^{a,b} and Kang Tai^{a,b,*}

^aCentre for Advanced Numerical Engineering Simulations,
School of Mechanical and Production Engineering,

Nanyang Technological University, 50 Nanyang Ave, Singapore 639798

^bSingapore-MIT Alliance, 4 Engineering Drive 3, Singapore 117576

*E-mail: mktai@ntu.edu.sg

Telephone: +65 67904444

Fax: +65 67911859

Abstract— In this study, an improved bit-array representation method for structural topology optimization using the Genetic Algorithm (GA) is proposed. The issue of representation degeneracy is fully addressed and the importance of structural connectivity in a design is further emphasized. To evaluate the constrained objective function, Deb's constraint handling approach is further developed to ensure that feasible individuals are always better than infeasible ones in the population to improve the efficiency of the GA. A hierarchical violation penalty method is proposed to drive the GA search towards the topologies with higher structural performance, less unusable material and fewer separate objects in the design domain in a hierarchical manner. Numerical results of structural topology optimization problems of minimum weight and minimum compliance designs show the success of this novel bit-array representation method and suggest that the GA performance can be significantly improved by handling the design connectivity properly.

I. INTRODUCTION

Structural optimization deals with optimal design of load-bearing structures. Structural topology optimization is one of the most important subset approaches of structural optimization which aims to find the best possible structure that meets different multidisciplinary requirements such as functionality and manufacturing [1]. Generally, structural topology optimization is a powerful tool which can help the designer select suitable initial structural topologies for conceptual design and more importantly, it is identified as economically the most rewarding task in structural design because it can achieve far greater savings than conventional (sizing or shape) optimization [2]–[4].

Structural topology optimization has yet been identified as one of the most challenging tasks in structural design. Various optimization methods have been developed over the past fifteen years since the pioneer work of Bendsøe and Kikuchi [3] and it is well recognized that structural topology optimization as a generalized shape optimization problem has received wide attention and experienced considerable progress recently, as reviewed in detail by Rozvany [2]. Those extensively developed methods can be classified as a few families

of structural topology optimization methods [5], [6]. One of the most established families of methods is the one based on the homogenization approach proposed by Bendsøe and Kikuchi [3], in which the structural form is represented by a sponge-like material with micro-scale cells with voids and the material throughout the structure is redistributed by using an optimality criteria procedure. Moreover, as an important alternative approach of this family of homogenization methods, the power-law approach [7], which is also called the SIMP (Solid Isotropic Microstructure with Penalization) method in [8] and originally introduced by Bendsøe in 1989 [9], has got a fairly general acceptance during the past few years [2]. Another important family of structural optimization methods is based on the Evolutionary Structural Optimization (ESO) method proposed by Xie and Steven [4], in which the material in a design domain which is not structurally active is considered as inefficiently used and can thus be removed by using some element rejection criteria. Both the homogenization method and the ESO method have been further developed by a large number of researchers and two different families of methods have thus been established [2], [5], [6]. It should be pointed out that although computationally effective, both cannot perform a global search and thus do not necessarily converge to the global optimal solution for the given objective function and constraints [2]. Another emerging family of structural topology optimization methods is the one using the Genetic Algorithms (GAs), which are gradually recognized as a kind of powerful and robust stochastic global search method [2]. More recently, GAs have been increasingly employed in the structural topology optimization field [6], [10]–[20].

It is well known that for the GAs, the choice of a representation method is of vital importance. Currently, the bit-array or binary-string representation method has been widely adopted [10]–[14]. The bit-array representation method [11], [15], which is similar to the binary-string representation method adopted by Jakiela and his co-workers [10], [13], as well as Fanjoy and Crossley [14], is an intuitive and straightforward

method. However, the design connectivity of each individual cannot be guaranteed since each element can be either solid or void and extra constraint on the connectivity is not involved in the bit-array representation itself. This issue can even lead to the failure of the GA to find a feasible solution, as reported in [15]. Another representation method is the Voronoi-based representation [11], [15], in which a finite number of Voronoi sites being labeled 0 or 1 are used to define the Voronoi diagram and to represent a partition of the design domain into two subsets. Although good results on the topological optimum design problems have been obtained [11], [15], the problem of design connectivity still exists in this representation method.

It has been well recognized that design connectivity is an important issue for the structural topology design [14]. However, this issue of design connectivity has to be fully addressed yet. Although most of the bit-array or binary string representation GAs have included the connectivity analysis [10], [11], [13]–[15], those design connectivity handling approaches cannot well bias the GA search to favor the formation of connected topologies from randomly generated disconnected topologies through evolution. Furthermore, they may cause the problem of representation degeneracy, analogous to recessive genes [14], which is acknowledged to be generally bad for the GA optimization [11].

The objective of the present study is to further address the design connectivity issue in the structural topology optimization using the bit-array representation GA. The number of connected objects is explicitly used as an equality constraint function and a hierarchical violation penalty method is proposed. An identical initialization method is also proposed. Numerical results based on the minimum compliance and minimum weight design problems demonstrate the good performance of the present GA in terms of accuracy and computational cost.

II. IMPLEMENTATION OF THE GA WITH A BIT-ARRAY REPRESENTATION METHOD

A. Chromosome Representation

In this study, the bit-array representation is adopted as the population chromosome representation method to define the distribution of material and void in a two-dimensional topology design domain, as shown in Figure 1, in which ‘1’ represents material and ‘0’ void. This is a straightforward and natural representation method [14] and the decoding step is quite simple and intuitive. However, disconnected structures, as shown in Figure 2, and checkerboard patterns, as shown in [13], which are not desirable in the practical applications, are also included in this representation. To improve the GA performance, additional strategies must be taken to bias the GA search towards connected structures and feasible individuals during the GA evolution.

B. Objective Function

In the field of structural topology optimization, the importance of the design connectivity should be highlighted. In the present study, different from all the previous work dealing

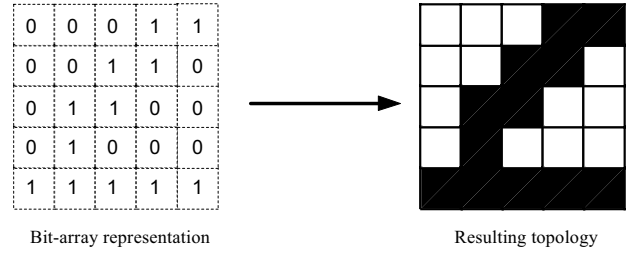


Fig. 1: Decoding step of the bit-array representation

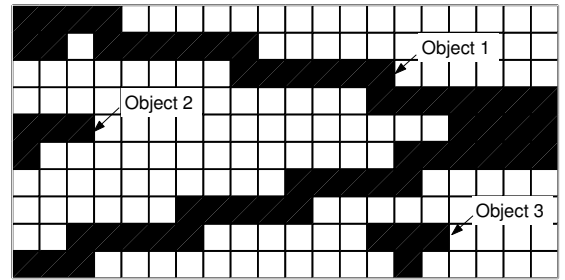


Fig. 2: Connected objects in a design domain

with the design connectivity issue [10], [11], [13]–[15], the number of connected objects in the design domain is explicitly adopted as an equality constraint function. Since all the other constraints such as the constraints on the volume fraction or the maximum displacement may be taken as inequality constraints, the general single-objective constrained optimization problem can be written as

$$\begin{aligned} \text{Minimize: } & f(\mathbf{x}), \mathbf{x} \in \mathbb{R} \\ \text{subject to } & g_i(\mathbf{x}) \leq 0, i = 1, 2, \dots, I \\ & n_c - 1 = 0 \end{aligned} \quad (1)$$

where \mathbf{x} is the solution matrix (bit-array) in the design domain \mathbb{R} , $f(\mathbf{x})$ the objective function, $g_i(\mathbf{x})$ the i -th inequality constraint function, I the number of inequality constraints, n_c is the number of connected objects in \mathbb{R} , as shown in Figure 2, in which 3 connected objects can be found.

This constrained optimization problem is next converted into an artificial unconstrained optimization problem by adopting Deb’s constraint handling approach [21], which can be written as

$$F(\mathbf{x}) = \begin{cases} f(\mathbf{x}) & \text{if } \mathbf{x} \in \mathbb{F} \\ \tilde{f} + \text{viol}(\mathbf{x}) & \text{otherwise} \end{cases} \quad (2)$$

where $F(\mathbf{x})$ is the artificial unconstrained objective function, \mathbb{F} the feasible region of the design domain \mathbb{R} , \tilde{f} the objective function value of the worst feasible solution in the population, and $\text{viol}(\mathbf{x})$ the summation of all the violated constraint function values.

C. Hierarchical Violation Penalty Method

A hierarchical violation penalty method is proposed to bias the GA search towards a combination of connected structures with better performance, fewer connected objects and less

unusable material in the design domain. Here a connected object is referred to as a group of elements represented in the regular mesh in which each element shares at least one element face (edge) with all the other elements of the group directly or indirectly through other elements. Thus, a connected object defined here is different from the connected elements defined in [10], [13] or connected structure defined in [14] or viable structure defined in [11], in which the support regions and loading regions must be connected properly. To transfer the loads to the support, the loading regions and the support regions must be properly connected. Furthermore, if there are more than one connected objects, it is impossible to transfer loads between those objects [14].

The basic idea of the proposed hierarchical violation penalty method is to drive the GA search towards connected structures and better performance based on different penalty multipliers. Multiple objects are driven into the formation of a single object by penalizing both the number of objects and the area of unusable objects. Single objects are driven into the formation of the ones with better performance.

According to the hierarchical violation penalty method, if the number of connected objects of the topology in the design domain is more than 1, the summation of all the violation values $\text{viol}(\mathbf{x})$ can be formulated as

$$\text{viol}(\mathbf{x}) = \Gamma_c (n_c - 1) + \Gamma_a \tilde{A}, \quad (3)$$

where Γ_c is the penalty multiplier for the number of connected objects of the topology in the design domain to drive the search towards fewer connected objects and Γ_a is the penalty multiplier for the minimum area \tilde{A} (in terms of number of objects) of all the disconnected objects, rather than total area of all the unusable objects that do not connect the loading regions and the support regions together [11], to favor the occurrence of load bearing objects. The values of these two penalty multipliers are problem dependent, but the former should be much larger than the latter to ensure that the number of connected objects is more heavily penalized. In the present numerical implementations, it is adopted that $\Gamma_c = A_0$, in which A_0 is the area of the whole design domain in terms of number of elements, and $\Gamma_a = 1$.

According to the hierarchical violation penalty method, if the number of connected objects of the topology in the design domain is equal to 1, different penalties will be applied according to the violation on the loading bearing ability and the violation on the inequality constraint functions. If this single connected object in the design domain connects both the support regions and the loading regions together, then the connected object turns out to be a connected structure and thus the finite element analysis can be performed. Based on the results of the finite element analysis, if there is any violation of the inequality constraints, the summation of all the violation values $\text{viol}(\mathbf{x})$ will be formulated as

$$\text{viol}(\mathbf{x}) = \Gamma_n \sum_i g_i(\mathbf{x}), \quad (4)$$

in which $g_i(\mathbf{x})$ is normalized by formulating it to have a

maximum violation of 1 [22], and the problem dependent penalty multiplier Γ_n is assigned as $\Gamma_n = 1$ in the present numerical implementation to favor the occurrence of connected structures. However, if the resulting single connected object of the topology in the design domain does not connect both the support regions and the loading regions properly, then the connected object turns out not to be a load bearing structure and thus the finite element analysis need not be performed. In this case, $\text{viol}(\mathbf{x})$ will be given a problem dependent prescribed worst value to obtain a near zero fitness to help eliminate such individuals in the population. In the present numerical implementation, it is adopted that $\text{viol}(\mathbf{x}) = (A_0)^2$.

Therefore, this hierarchical violation penalty method can drive the genetic algorithm search towards topologies with either higher structural performance or fewer connected objects in the design domain or less unusable material. Furthermore, the problem of representation degeneracy has been fully accounted for by the hierarchical penalty.

D. Rank-Based Fitness Assignment

Rank-based fitness assignment usually behaves in a robust manner. In the present study, Baker's linear ranking algorithm [23] with a selective pressure of 2 is used. The fitness of each individual in the population is defined as

$$\hat{F}(\mathbf{x}_i) = \frac{2(n_i - 1)}{N_{\text{ind}} - 1}, \quad (5)$$

where $\hat{F}(\mathbf{x}_i)$ is the fitness of the i -th individual, n_i the position of the i -th individual in the individuals rank, and N_{ind} the population size.

E. Population Initialization

Usually, population initialization is achieved by generating the required number of individuals using a random number generator that uniformly distributes numbers in the desired range. However, by using this initialization method on the bit-array representation, it is often found that there is no connected structure in the initial population and the population in the early generations and the GA may thus fail to converge if the finite element mesh is not too coarse. To guarantee the existence of connected structures in the population for such problems, an alternative population initialization method is also proposed. In this method, all individuals in the initial population are identical and they are designs where the whole design domain is filled with material (i.e. there are no void elements). With appropriately selected GA operators, the convergence of this GA can also be achieved since the diversity of the population in the early generations can be achieved mainly through mutation operation.

F. Other GA Operators

Other GA operators include the SUS (Stochastic Universal Sampling) selection method, the uniform crossover method, binary mutation, and an elitist strategy to ensure that the most fit individuals in the population propagate through successive generations.

III. RESULTS AND DISCUSSION

With those introduced GA operators, the proposed bit-array representation GA is applied to evolutionary structural topology design problems. Unless otherwise stated, the following settings are used in the numerical experiments presented below: standard SGA evolution with a population size of 100 and a generation gap (the fractional difference between the new and old population sizes [24]) of 0.9 and a mutation rate of 0.01; all runs are stopped after 500 generations; all the runs are carried out using MATLAB; all the number of function evaluations are referred to the number of the finite element analysis based on a connected structure with a single connected object; and all the FE function evaluations are based on Sigmund's corresponding FEA code in [7], in which a planar stress square element is used; the results obtained for each problem are based on 20 independent runs of the SGA; and all the CPU time is based on a desktop computer with an Intel Pentium IV processor of 2 GHz. As to the physical property of the plates, it is also assumed that $Et = 1$, where E is Young's elasticity modulus and t the thickness of the plate, and Poisson's ratio $\nu = 0.3$.

A. Minimum Compliance Design

The example presented here is a minimum compliance optimal topology design problem [7], which can be expressed as

$$\begin{aligned} \text{Minimize: } & C(\mathbf{x}), \mathbf{x} \in \mathbb{R} \\ \text{subject to } & V(\mathbf{x})/V_0 \leq f \end{aligned} \quad (6)$$

where \mathbf{x} is a solution vector, $C(\mathbf{x})$ the compliance of the topology, $V(\mathbf{x})$ and V_0 the material volume and the design domain volume, respectively, and f the prescribed volume fraction.

This example will demonstrate the performance of the present GA in dealing with structural topology optimization of a clamped beam with a high volume fraction ($f = 0.7$), which can be classified as a generalized shape optimization problem [2] and usually FE-based topology optimization algorithms should be involved. As shown in Figure 3, a symmetric design domain of the size $2H \times H$ ($H = 5$) discretized into a 24×12 mesh is used and the loading of $P = 1$ is applied. This problem is similar to the one in the standard topology optimization literature (see [25]).

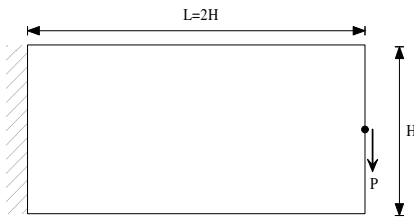


Fig. 3: Design domain of a 2×1 clamped beam with a volume fraction of 0.7.

The result of Figure 4 is obtained by use of the 99 line MATLAB code [7] of Sigmund's power-law approach

together with a filtering technique for controlling geometric complexity and ensuring existence of solutions to the topology optimization problem (6). It can be seen that the boundary of the geometry is quite fuzzy. Since the power-law approach is a continuous material density re-distribution of the finite elements in the design domain, integrating the result of Figure 4 into an automated structural design procedure would be a major difficulty. Usually, it is left to the designer to use a particular approach to finish the post-processing task of the fuzzy topology [26]. Figure 5 displays that better geometry can also be obtained by using a finer mesh of 60×30 , however, the computational cost would also increase more significantly (204 times larger in CPU time) than the improvement in the compliance value (3.69% better).

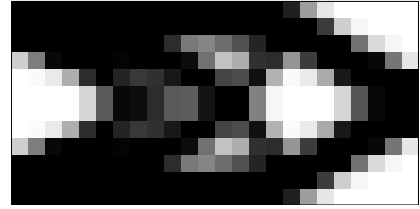


Fig. 4: Optimal topology obtained by the power-law approach [7], $C = 50.37$ (24×12 mesh).

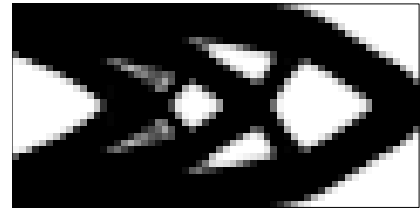


Fig. 5: Optimal topology obtained by the power-law approach [7], $C = 48.51$ (60×30 mesh).

The result of Figure 6 is the best topology obtained by use of the present bit-array representation GA together with a hierarchical violation penalty method to bias the GA search towards a combination of connected structures with better performance, fewer connected objects and less unusable material in the design domain. It can be seen that the boundary of the optimal geometry is quite distinctive (black and white design) and the optimal compliance is better than those of the previous two. The success of the present bit-array representation method together with the GA is thus demonstrated. Furthermore, the efficiency of adopting Schoenauer's penalization approach [11] on the area of the disconnected objects, in which the total area of the unusable objects is used, in the present hierarchical violation penalty method is also studied. The corresponding best topology is shown in Figure 7. It can be seen that good result can also be obtained with a bit worse compliance value (0.91% higher). The result of Figure 8 also suggests that the penalization on the minimum area, rather than the total area, of the disconnected objects would behave better, which may

contribute to the higher efficiency of the former in reducing the total number of disconnected objects.

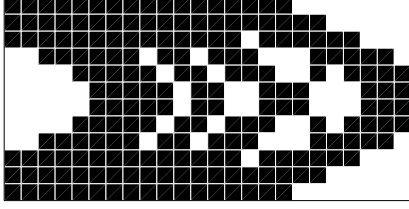


Fig. 6: Optimal topology obtained by the present GA, $C = 47.14$ (minimum area penalization).

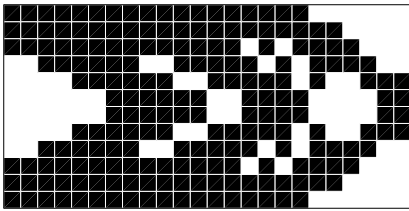


Fig. 7: Optimal topology obtained by the present GA, $C = 47.57$ (total unusable area penalization).

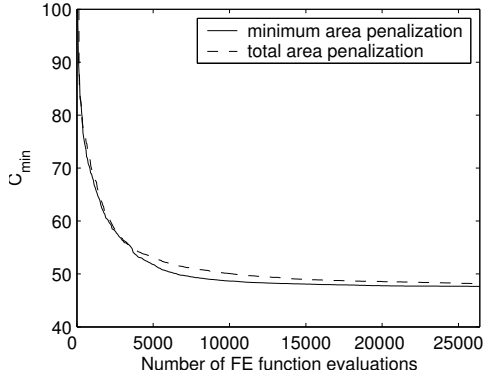


Fig. 8: Convergence of the bit-array representation GA (averaged over 20 runs).

B. Minimum Weight Design

The minimum weight optimal topology design problem can be expressed as

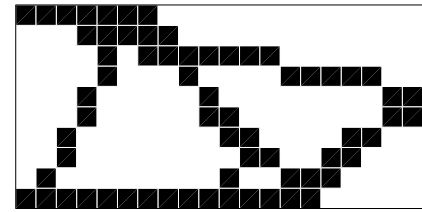
$$\begin{aligned} & \text{Minimize: } W(\mathbf{x}), \mathbf{x} \in \mathbb{R} \\ & \text{subject to } D(\mathbf{x})_{\max}^i - D_{\lim}^i \leq 0, i = 1, 2, \dots, J \\ & \quad n_c - 1 = 0 \end{aligned} \quad (7)$$

where $W(\mathbf{x})$ the weight of the topology, $D(\mathbf{x})_{\max}^i$ the maximal displacement under loading i , D_{\lim}^i the prescribed displacement limit of loading i and J the total number of load cases. In the present study, only one load case is considered.

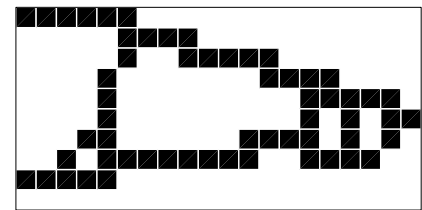
The minimum weight optimal topology design problem in [15] is adopted in this study to compare the results of the

present bit-array representation with those of the Voronoi-bar representation presented in [15], in which each individual is a list of Voronoi-bars. A cantilever plate of aspect ratio 2×1 with the left boundary fixed and a unit point force applied vertically downward at half-height of the right boundary is studied. The plate is discretized into a 20×10 mesh, and the limit on the maximal displacement is $D_{\lim}^i = 220$.

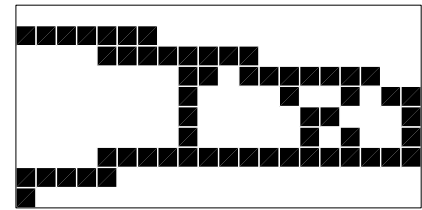
The minimum weight design of the 2×1 plate using the present GA is carried out. For this instance, the average CPU time of one generation is 4.1 s. Figure 9 displays results of those best topologies of the 2×1 plate. It can be seen that the two best topologies given by the present bit-array representation are with lower weight values 0.29 and 0.295, respectively, and these are corresponding improvements of 12.1% and 10.6%, compared to that of the best topology given by the Voronoi-bar representation [15] (with a higher weight value of 0.33). Figure 10 shows the best convergence characteristics of the present bit-array representation GA. As expected, the GA with random initial population converges faster than that with identical initial population due to the relatively higher diversity produced by the former. Hence, it is demonstrated that the present bit-array representation outperforms the Voronoi-bar representation in accuracy and computational cost for the 2×1 plate while using the GAs.



(a) Voronoi-bar [15], $W(\mathbf{x}) = 0.33$



(b) Bit-array (with random initial population), $W(\mathbf{x}) = 0.29, D_{\max} = 219.48$



(c) Bit-array (identical initial population), $W(\mathbf{x}) = 0.295, D_{\max} = 216.22$.

Fig. 9: Comparison of the best topologies of the 2×1 plate.

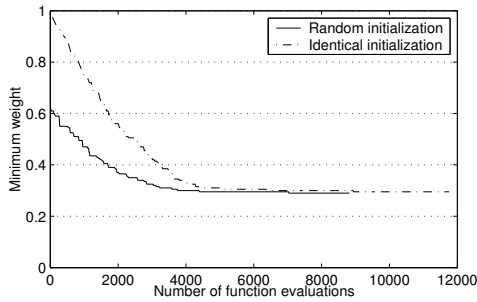


Fig. 10: Convergence of the bit-array representation GA on the 2×1 plate (of the best GA run).

IV. CONCLUSION

A high performance bit-array representation method for structural topology optimization using the GA is proposed. The importance of design connectivity is further emphasized by using the number of connected objects explicitly as an equality constraint function and the issue of representation degeneracy is fully addressed. A hierarchical violation penalty method is proposed to penalize the violated constraint functions in a hierarchical way to eliminate the representation degeneracy and drive the GA search towards the combination of better structural performance, fewer connected objects and less unusable material in the design domain. The problem of representation degeneracy is thus alleviated and the GA convergence to connected structures can be well guided. With appropriately selected GA operators, the proposed bit-array representation GA is applied into the minimum weight design problems. Numerical results show that the present bit-array representation method GA outperforms the popular power-law approach in the optimal compliance value and the quality of the geometry boundary due to its effective discrete global search capacity, furthermore, in GAs the present bit-array representation outperforms the Voronoi-bar representation method in terms of accuracy and computational cost due to its significant advantage in evolving into load bearing structures.

ACKNOWLEDGMENT

Part of this work was supported by the Singapore Ministry of Education Academic Research Fund through research grant RG92/98, which the authors gratefully acknowledge.

REFERENCES

- [1] J. E. Akin and J. Arjona-Baez, "Enhancing structural topology optimization," *Engineering Computations*, vol. 18, no. 3-4, pp. 663–675, 2001.
- [2] G. I. N. Rozvany, "Aims, scope, methods, history and unified terminology of computer-aided topology optimization in structural mechanics," *Structural and Multidisciplinary Optimization*, vol. 21, no. 2, pp. 90–108, 2001.
- [3] M. P. Bendsøe and N. Kikuchi, "Generating optimal topologies in structural design using a homogenization method," *Computer Methods in Applied Mechanics and Engineering*, vol. 71, pp. 197–224, 1988.
- [4] Y. M. Xie and G. P. Steven, "A simple evolutionary procedure for structural optimization," *Computers and Structures*, vol. 49, no. 5, pp. 885–896, 1993.

- [5] S. Bulman, J. Siens, and E. Hinton, "Comparisons between algorithms for structural topology optimization using a series of benchmark studies," *Computers and Structures*, vol. 79, no. 12, pp. 1203–1218, 2001.
- [6] S. Y. Wang and K. Tai, "Structural topology design optimization using genetic algorithms with a bit-array representation," *Computer Methods in Applied Mechanics and Engineering*, 2003, submitted for review.
- [7] O. Sigmund, "A 99 line topology optimization code written in MATLAB," *Structural and Multidisciplinary Optimization*, vol. 21, no. 2, pp. 120–127, 2001.
- [8] G. I. N. Rozvany, M. Zhou, and T. Birker, "Generalized shape optimization without homogenization," *Structural Optimization*, vol. 4, pp. 250–254, 1992.
- [9] M. P. Bendsøe, "Optimal shape design as a material distribution problem," *Structural Optimization*, vol. 1, pp. 193–202, 1989.
- [10] D. Chapman, K. Saitou, and M. J. Jakiela, "Genetic Algorithms as an approach to configuration and topology design," *Journal of Mechanical Design*, vol. 116, no. 4, pp. 1005–1012, 1994.
- [11] M. Schoenauer, "Shape representation for evolutionary optimization and identification in structural mechanics," in *Genetic Algorithms in Engineering and Computer Science*, G. Winter, J. Periaux, M. Galan, and P. Cuesta, Eds. Chichester: John Wiley, 1995, pp. 443–464.
- [12] C. Kane and M. Schoenauer, "Topological optimum design using Genetic Algorithms," *Control and Cybernetics*, vol. 25, no. 5, pp. 1059–1088, 1996.
- [13] M. J. Jakiela, C. Chapman, J. Duda, A. Adewuya, and K. Saitou, "Continuum structural topology design with Genetic Algorithms," *Computer Methods in Applied Mechanics and Engineering*, vol. 186, no. 2-4, pp. 339–356, 2000.
- [14] D. W. Fanjoy and W. A. Crossley, "Topology design of planar cross-sections with a Genetic Algorithm: Part 1—Overcoming the obstacles," *Engineering Optimization*, vol. 34, no. 1, pp. 1–12, 2002.
- [15] H. Hamda, F. Jouve, E. Lutton, M. Schoenauer, and M. Sebag, "Compact unstructured representations for evolutionary design," *Applied Intelligence*, vol. 16, no. 2, pp. 139–155, 2002.
- [16] S. Y. Wang and K. Tai, "Graph representation for evolutionary structural topology optimization," *Computers & Structures*, 2003, submitted for review.
- [17] K. Tai and T. H. Chee, "Genetic Algorithm with structural morphology representation for topology design optimization," in *Mechanics in Design—Proceedings of the 2nd International Conference on Mechanics in Design*. Nottingham, U. K., 1998, pp. 827–836.
- [18] —, "Design of structures and compliant mechanisms by evolutionary optimization of morphological representations of topology," *ASME Journal of Mechanical Design*, vol. 122, no. 4, pp. 560–566, 2000.
- [19] K. Tai, G. Y. Cui, and T. Ray, "Design synthesis of path generating compliant mechanisms by evolutionary optimization of topology and shape," *ASME Journal of Mechanical Design*, vol. 124, no. 3, pp. 492–500, 2002.
- [20] S. Akhtar, K. Tai, and J. Prasad, "Topology optimization of compliant mechanisms using evolutionary algorithm with design geometry encoded as a graph," in *Proceedings of the ASME International Design Engineering Technical Conferences*, Montreal, Canada, 2002, paper no. DETC2002/DAC-34147.
- [21] K. Deb, "An efficient constraint handling method for Genetic Algorithms," *Computer Methods in Applied Mechanics and Engineering*, vol. 186, no. 2-4, pp. 311–338, 2000.
- [22] —, *Optimization for engineering design: Algorithms and examples*. New Delhi: Prentice-Hall, 1995.
- [23] J. E. Baker, "Adaptive selection methods for Genetic Algorithms," in *Proceedings of the International Conference on Genetic Algorithms and Their Applications*, J. J. Grefenstette, Ed. Pittsburgh, PA, USA: Lawrence Erlbaum Associates, 1985, pp. 101–111.
- [24] K. A. D. Jong and J. Sarma, "Generation gaps revisited," in *Foundations of Genetic Algorithms 2*, L. D. Whitley, Ed. San Mateo: Morgan Kaufmann Publishers, 1993, pp. 19–28.
- [25] M. P. Bendsøe, A. R. Diaz, and N. Kikuchi, "Topology and generalized layout optimization of elastic structures," in *Topology Design of Structures*, ser. NATO ASI Series E, M. P. Bendsøe and C. A. M. Soares, Eds. Dordrecht: Kluwer Academic Publishers, 1993, vol. 227, pp. 159–206.
- [26] T. E. Bruns and D. A. Tortorelli, "Topology optimization of non-linear elastic structures and compliant mechanisms," *Computer methods in Applied Mechanics and Engineering*, vol. 190, no. 26-27, pp. 3443–3459, 2001.