Struct Multidisc Optim (2005)
DOI 10.1007/s00158-004-0498-5

RESEARCH PAPER

I.Y. Kim · O.L. de Weck

Variable chromosome length genetic algorithm for progressive refinement in topology optimization*

Received: 5 April 2004 / Revised manuscript received: 16 September 2004 / Published online: 7 January 2005

Abstract This article introduces variable chromosome lengths (VCL) in the context of a genetic algorithm (GA). This concept is applied to structural topology optimization but is also suitable to a broader class of design problems. In traditional genetic algorithms, the chromosome length is determined a priori when the phenotype is encoded into the corresponding genotype. Subsequently, the chromosome length does not change. This approach does not effectively solve problems with large numbers of design variables in complex design spaces such as those encountered in structural topology optimization. We propose an alternative approach based on a progressive refinement strategy, where a GA starts with a short chromosome and first finds an 'optimum' solution in the simple design space. The 'optimum' solutions are then transferred to the following stages with longer chromosomes, while maintaining diversity in the population. Progressively refined solutions are obtained in subsequent stages. A strain energy filter is used in order to filter out inefficiently used design cells such as protrusions or isolated islands. The variable chromosome length genetic algorithm (VCL-GA) is applied to two structural topology optimization problems: a short cantilever and a bridge problem. The performance of the method is compared to a brute-force approach GA, which operates ab initio at the highest level of resolution.

Keywords Consanguineous · Genetic algorithms · Multistage search · Progressive refinement · Structural topology optimization · Variable chromosome length

* Presented as paper AIAA–2004–1911 at the 45th AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference, Palm Springs, California, April 19–22, 2004

I.Y. Kim** · O.L. de Weck (✉)
Department of Aeronautics & Astronautics, Engineering Systems Division, Massachusetts Institute of Technology, Cambridge, Massachusetts 02139, USA
E-mail: iykim@me.queensu.ca, deweck@mit.edu
** *Present address:* Dept. of Mechanical Engineering, Queen's University, Kingston, Ontario, K7L 3N6, Canada

1 Introduction

Genetic algorithms (GA) have gained increasing popularity in design optimization, operations research, and for general combinatorial search problems (Holland 1975; Goldberg 1989). A key operator is the encoding step which translates the design variables from the phenotypic space to the genotypic space, resulting — for example — in a binary, real, or hexadecimal chromosome. The length of the chromosome, i.e. the number of alleles, in the genotype space is a surrogate for the amount of information describing an artifact in the phenotype space. Figure 1 shows examples of both the phenotype and genotype (chromosomes) of two hypothetical designs.

The chromosome length, L, in units of bits is a function of the number of design variables, $x_i \in \mathbb{R}$, where $i = 1, 2, \ldots, n$, the resolution level of each variable, Δx_i, as well as the dimensionality, d, of the encoding base. For binary encoding we set $d = 2$. One can then estimate the chromosome length as:

$$L = f(n, \Delta x_i, d) = \sum_{i=1}^{n} \left\lceil \ln\left(\frac{x_{i,UB} - x_{i,LB}}{\Delta x_i}\right) \Big/ \ln d \right\rceil \quad (1)$$

where $x_{i,UB}$ and $x_{i,LB}$ are the upper and lower bounds on the i-th design variable, respectively, and $\lceil\ \rceil$ is the ceiling function (rounding to the next largest integer).

Traditionally, in genetic design optimization the chromosome length is fixed a priori and it cannot change in subsequent generations. Evolution in this sense is understood as the process of approaching an optimal instantiation of alleles given a constant phenotype-genotype mapping and a fixed chromosome length. We suggest that this traditional approach has some disadvantages when dealing with complex design problems. First, the best achievable fitness is inherently limited by the chromosome length. Hence, the fitness asymptote that is typically observed in genetic optimization is as much a result of constraints as of the number of design variables and their resolutions. Second, the problem is that we do not a priori know how much design free-

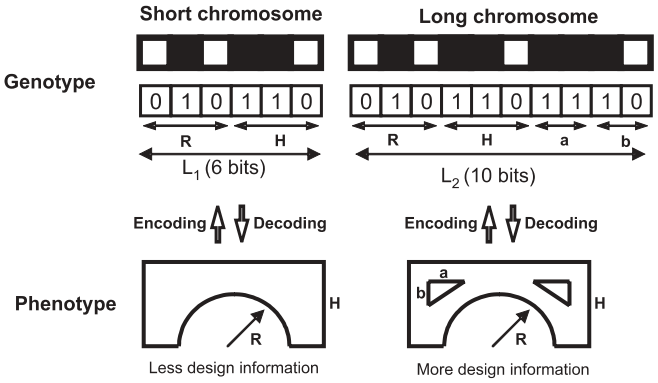


Fig. 1 Examples of designs in the phenotype and corresponding chromosomes in the genotype domain

dom is required and consequently how long chromosomes should be. If short chromosomes are used, one may not obtain good solutions due to the lack of design freedom. On the other hand, if chromosome length is excessive for a particular problem, it will cause a high computational burden without much performance benefit.

This article presents the development of an effective genetic algorithm that can change the chromosome length by implementing the design principle "from coarse to fine", otherwise referred to as *progressive refinement*. We hypothesize that significant fitness improvements can be achieved by gradually increasing chromosome length. We will also show that increasing the design freedom gradually by extending chromosome lengths in stages allows for the reduction of computational costs for complex problems with many design variables. The increase in chromosome length can be achieved either by an increase in the resolution of existing design variables or by the addition of new design variables during encoding.

1.1 Literature review

We will first review the general literature on structural topology optimization. This will be followed by a focused discussion of previous work in the area of variable bit-string genetic algorithms in the context of structural optimization. Michell (1904) first studied structural topology optimization and obtained an analytical solution, called Michell trusses, which have an infinite number of truss members. For structural shape optimization, Cea (1981), Zolésio (1981), Rousselet (1981) and Haug et al. (1986) developed the theory for analytical sensitivity analysis based on the continuum approach. Haftka and Grandhi (1986) and Kwak (1994) reviewed structural optimization methods and shape sensitivity analysis. Bendsøe and Kikuchi (1988) developed the homogenization method for topology optimization.

The evolutionary structural optimization method, in which the von Mises stress was adopted as the measure of performance of each cell in a structural member, was studied by Xie and Steven (1993) and Guan et al. (2003). Later the principal stress-based method was developed to deal with tension-dominated or compression-dominated material. Maute and Ramm (1995) proposed an adaptive topology and shape optimization method. They performed shape optimization and topology optimization separately and mapped the results to each other. DeRose and Diaz (2000) developed a meshless, wavelet-based layout optimization method. In order to overcome the problems of mesh degradation in convergence for large-scale layout optimization problems, a fictitious domain and a wavelet-Galerkin technique were used. Kim and Yoon (2000) proposed a multi-resolution topology optimization based on a multi-scale wavelet-Galerkin technique, and later, an adaptive multi-scale, wavelet-Galerkin method was developed by Diaz (1999), Kim et al. (2003).

Kim and Kwak (2002) proposed a generalized topology optimization formulation where the design domain grows in order to obtain better solutions that cannot be obtained by conventional methods. Diaz and Bendsøe (1992) dealt with problems with multiple loading conditions, and Min et al. (2000) studied multi-objective topology optimization considering static compliance and eigenvalues. Extensive reviews of topology optimization can also be found in Bendsøe (1995), Kirsch (1989) and Rozvany et al. (1995). More recent reviews can be found in Eschenauer and Olhoff (2001), Kim et al. (2002) and Mackerle (2003).

The drawbacks of gradient-based approaches or optimality criteria methods are that (1) they may converge to a local optimum and that (2) often intermediate densities are obtained, which are not physically meaningful. Genetic algorithms can remedy these problems effectively, albeit at additional computational expense. GAs explore the entire design space and are less likely to get trapped in local minima, provided that sufficiently large initial populations and mutation rates are used. Moreover, it is possible to have only binary values of density, ON (= 1) or OFF (= 0), at each cell. Chapman et al. (1994) used GAs for continuum topology optimization with domain refinement. However, the chromosome length did not change in their study. In order to create diversity, a high mutation rate was used. This work has been extended to several types of fitness functions (Chapman and Jakiela 1996; Jakiela et al. 2000).

Lin and Hajela (1993, 1994) as well as Hajela and Lee (1995) used genetic algorithms for large-scale problems and truss topology optimization problems. Ryoo and Hajela (2004) also developed a genetic algorithm for topology optimization that handles variable string lengths. This work allows crossover between chromosomes with different lengths. While Ryoo and Hajela also implemented an efficient micro GA technique in their context, they did not *actively control* the chromosome length in order to achieve progressive refinement. In Hajela's work, multiple chromosome lengths exist in each generation to allow crossover between chromosomes with different lengths, but the number of different chromosome lengths remains the same from the beginning until the end. While Hajela's method was applied to a truss structure and composite panel, the VCL GA presented here is applied to continuum structures with a large numbers of cells. Todoroki and Haftka (1997) come to a similar conclusion as we do in their development of a two-stage 'consanguineous' genetic algorithm. They applied their work to the problem of optimizing the stack-

ing sequence of a multi-layered composite material and found significant improvement in the GA's reliability when a coarse representation of the stacking sequence was optimized first and subsequently used to create the initial population for a second, finer stage. However, they restricted their work to two stages and the stacking problem.

The contribution of our work is threefold. First, we combine the concepts of variable chromosome lengths and multi-stage search to achieve progressive refinement. We demonstrate the superiority of this strategy compared to a 'brute-force' search starting from a finely resolved, but random initial population. Second, we demonstrate that the multi-stage approach also promotes the use of constraint relaxation and imposition during various stages of the search process. Third, we develop the concept of seeding the design spaces of finer resolution with mutated best designs from design spaces of coarser resolution.

1.2 Motivation

We observed that the solutions obtained by the GA topology optimization method developed by Chapman et al. (1994) were quite noisy. This is primarily caused by the combinatorial size of the search space with many design variables and the difficulty of filtering out bad designs. In addition, it was not possible to impose mass constraints to the optimization problem as formulated by Chapman et al. (1994).

The design space in topology optimization is typically very large, as can be seen in their work. Hence, the GA requires large population sizes and long computing times. For example, the combinatorial number of design choices is $2^{640} \approx 4.6 \times 10^{192}$ or larger for a simple two-dimensional cantilever optimization problem (32×20 cells). Among the hypothetical design choices, only a very small fraction ($\ll 5\%$) is physically meaningful, and it is not efficient to explore the entire design space. One of the main difficulties is producing new designs that are meaningful and that feature no disconnected material regions or load paths.

In this paper, we first develop a variable chromosome length GA using a multi-stage search strategy. This reduces the size of the search space for exploration of problems with many design variables. It also increases the likelihood of approaching the global optimum by gradually refining the design space. This procedure allows a gradual imposition of mass constraints. Second, we adopt *strain energy cell ranking* as a measure for filtering out inferior designs. During the multi-stage procedure a non-dominated (approximate Pareto) front is formed, revealing the tradeoff between structural mass and compliance.

2 Variable chromosome length genetic algorithm

The overall procedure for topology design optimization using the variable chromosome length genetic algorithm (VCL-GA) is shown in Fig. 2. The inner loop represents the typical topology optimization by GA. Strain energy ranking is used to ensure the connectivity of structural elements and to specify the mass constraint (Sect. 3). Optimization starts from a short chromosome length, and when it converges at one stage, it transitions to the next stage featuring longer chromosomes or higher refinement.

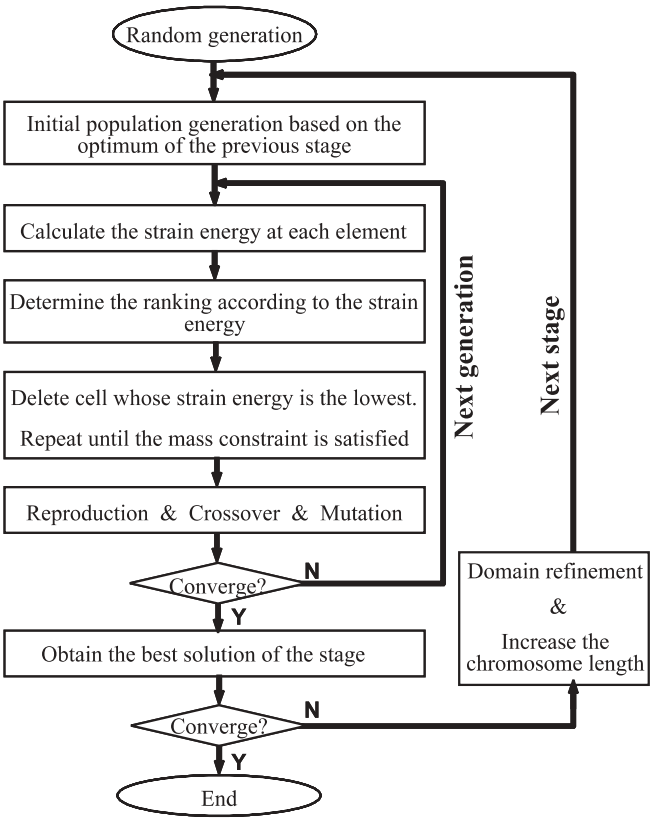


Fig. 2 Flow diagram of variable length chromosome genetic algorithm

Convergence within the inner loop is achieved when the rate of progress from one generation to the next, as measured by the average population fitness, falls below a numerical tolerance. Convergence of the outer loop is a more difficult question since this depends on the desirable resolution of the design. We use both the final mass constraint and a maximum chromosome length as termination criteria for the outer loop.

An example of topology optimization is shown in Fig. 3. We consider a short cantilever whose left side is clamped with a concentrated load applied at the mid-point of the right side. The design domain is discretized into rectangular cells. The normalized density of each cell becomes a design variable. Binary encoding is used and the density is either zero

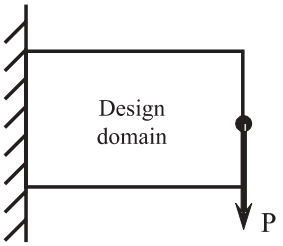


Fig. 3 Short cantilever problem

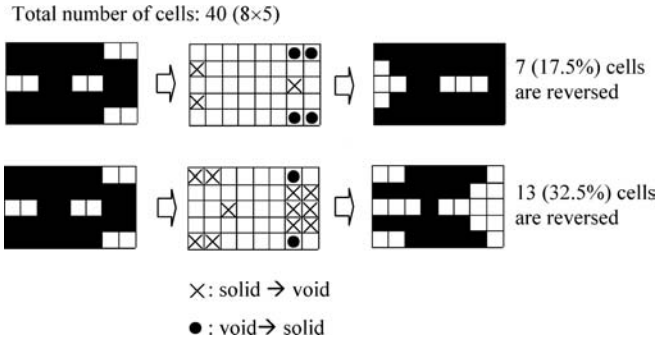


Fig. 4 Sample cases of generating mutants

(void) or one (solid), which leads to a binary 0-1 choice for each cell.

In the computational implementation of this problem, the elasticity of a void cell is set to a very low value. The objective function to be minimized is the compliance of the structure. The optimization problem statement is

$$\text{Minimize} \int_{\Omega} F^i z^i \, \mathrm{d}\Omega \,,$$
$$\text{Subject to} \int_{\Omega} \rho \, \mathrm{d}\Omega \leq M_0 \,,$$
$$0 \leq \rho \leq 1 \tag{2}$$

where F^i is i-th component of force vector $\mathbf{F}$, z^i is i-th component of displacement vector $\mathbf{z}$, Ω is the design domain, ρ is the normalized density of each cell, and M_0 is the maximum allowable mass which is often expressed as a percentage of the design domain volume. The design variables in this optimization are the densities of the cells.

In the proposed adaptive genetic algorithm, the design domain is initially discretized into coarse design cells, which are represented by short chromosomes in the early stages of design. The optimal solution of this stage is transferred to the next stage that has more refined design cells and therefore longer chromosomes. In this transition, the optimal solution is used to generate the initial population of the next stage. Some individuals of the next population are exact copies of the previous-stage optimum (elitism), and other members of the population are mutated variants stemming from the optimum. This seeding of the initial populations in finer stages is the main mechanism by which information is transferred between stages. This allows us to keep the number of generations and population sizes small compared to a brute-force approach, in which topology optimization would start at the finest resolution level with a random initial population. This brute-force approach is later used as a baseline for comparison (Fig. 12).

When making the mutants, the density of each cell is determined by the following equation:

$$\rho_k^{\text{New}} = \text{round}\left(\alpha + \beta\rho_k^{\text{Previous}} + \gamma \cdot \Gamma\right) \tag{3}$$

where ρ_k^{New} is the density of the k-th cell in the new stage, ρ_k^{Previous} is the density of the k-th cell in the previous stage, Γ is a uniformly distributed random variable on the interval [0,1], and "round( )" is the function that rounds off to the nearest integer. In this paper, the values of the parameters, α, β, and γ are set 0.1, 0.3 and 0.5, respectively.

With these values, the probability of retaining the density of the previous stage in an individual cell is 80%, and the probability of reversing it is 20%. By adjusting the three pa-

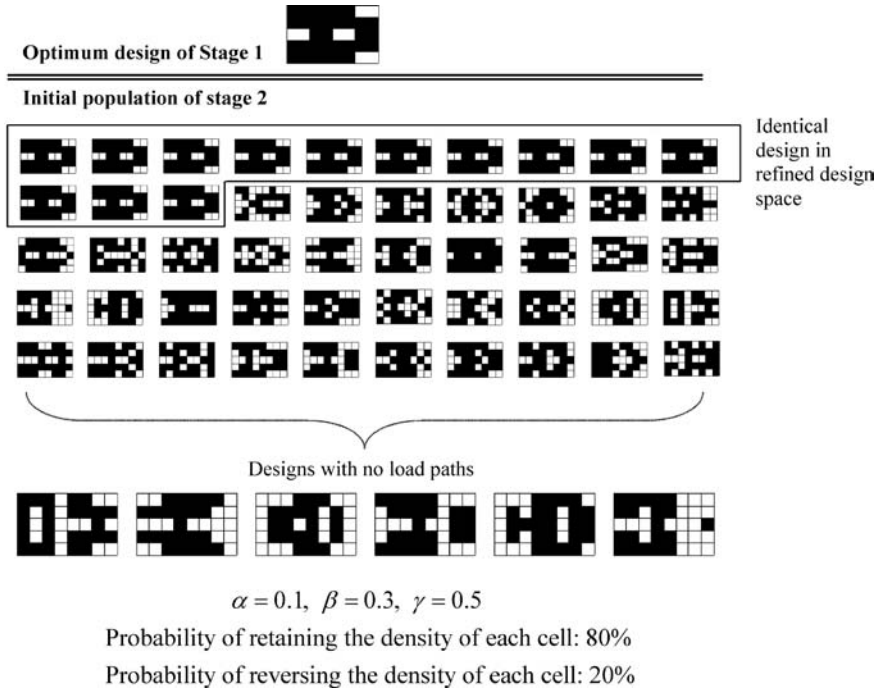


Fig. 5 Initial population of the second stage. Six designs do not have closed-load paths

rameters, the degree of diversity generated can be controlled, depending on the characteristics of the problem. Figure 4 shows two sample cases for generating mutants, in which 17.5% and 32.5% of the cells are reversed, respectively. This mechanism makes it possible to transfer a configuration to a stage with a different chromosome length, while maintaining diversity in the population.

Figure 5 represents an example of an optimum of the first stage and 50 individuals of the initial population of the second stage generated with this scheme. The optimum design in the first stage is transferred to the second stage by elitism and by the mutation equation (3). The first 13 designs are exactly the same, but they are mapped to a more-refined design domain. Another feature of this operation is that designs with open load paths may be generated. In this example, six designs do not have closed load paths. They are infeasible designs because theoretically the compliance is infinite. They serve only as diversity-generating seeds in the process of crossover, and later they are rejected from the population because of their low fitness. In this algorithm, the elitism GA replaces the worst designs with open-load paths with the best design at the end of each generation.

Table 1 Strain energy and cell ranking (protrusion case)

Element	Strain energy	Ranking	Feature
1	9.4419	3	
2	8.0666	4	
3	9.4815	2	
4	4.9600	6	
5	3.3997×10^{-17}	18	*Void*
6	6.1725×10^{-1}	12	**Protrusion**
7	1.2187×10^{-18}	24	*Void*
8	6.4800×10^{-18}	21	*Void*
9	2.0019×10^{-17}	20	*Void*
10	3.8160×10^{-1}	13	**Protrusion**
11	2.8523×10^{-17}	19	*Void*
12	5.0708	5	
13	9.8526	1	
14	2.5776	9	
15	2.8597	8	
16	3.7503×10^{-17}	17	*Void*
17	6.9412×10^{-17}	16	*Void*
18	8.9555×10^{-17}	15	*Void*
19	9.5157×10^{-17}	14	*Void*
20	2.6762×10^{-18}	23	*Void*
21	4.7923×10^{-18}	22	*Void*
22	1.0727	11	
23	1.7719	10	
24	4.6901	7	

3 Strain energy filter

One of the difficulties in applying GAs to structural topology optimization is that noisy designs are often produced. In particular, designs with protrusions or islands are inferior solutions because these protrusions or islands contribute very little or nothing to the strength of a structure. A technique for filtering out protrusions was considered in previous research (Chapman and Jakiela 1996). However, it was applicable only for a protrusion with a corner point connection, and a protrusion with an edge connection or an island was not dealt with.

In this work, we use strain energy as the contribution measure of each cell for GA-based topology optimization:

$$U_k = \frac{1}{2}\int_{\Gamma_k} \varepsilon_k \, \sigma_k \, \mathrm{d}\Gamma = \frac{1}{2}\int_{\Gamma_k} \varepsilon_k^T \mathbf{D}_k \varepsilon_k \, \mathrm{d}\Gamma \tag{4}$$

where U_k is the strain energy in the k-th cell, Γ_k is the domain of the k-th cell, ε_k is the strain tensor of the k-th cell, σ_k is the stress tensor of the k-th cell, and $\mathbf{D}_k$ is the elasticity tensor of the cell. In each step, the strain energy of each cell is computed, and then the ranking of the cells in terms of their strain energy is determined. Figure 6 demonstrates the filtering procedure for noisy designs with protrusions and islands in the short cantilever example. The figures in the second column show the strain energy distribution in each case.

Tables 1 and 2 present the strain energy values and the ranking for each case shown in Fig. 6. The cells are numbered according to Fig. 7.

Void cells have strain energies that are almost zero, and cells whose strain energies are relatively low are protrusions or isolated islands to be deleted by the filter. This method determines the ranking of the cells according to their strain energy and then rejects them one by one beginning from the last ranked until the mass constraint is met.

This scheme enables the mass constraint imposition at an arbitrary level in any stage. However, our experience re-

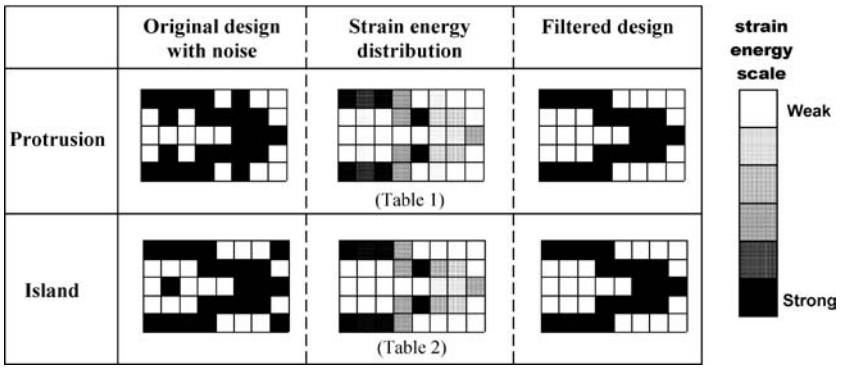


Fig. 6 Strain energy filter for removing protrusions and islands

Table 2 Strain energy and cell ranking (island case)

Element	Strain energy	Ranking	Feature
1	1.0049×10^{1}	2	
2	9.3191	4	
3	9.784	3	
4	4.9424	6	
5	3.3772×10^{-17}	14	*Void*
6	1.8232×10^{-17}	16	*Void*
7	2.2150×10^{-18}	23	*Void*
8	1.2859×10^{-5}	12	**Protrusion**
9	1.2928×10^{-17}	19	*Void*
10	1.4381×10^{-17}	18	*Void*
11	1.5363×10^{-17}	17	*Void*
12	4.9748	5	
13	1.0109×10^{1}	1	
14	4.4807	8	
15	2.9549	9	
16	2.8325×10^{-17}	15	*Void*
17	7.1200×10^{-18}	21	*Void*
18	6.0905×10^{-6}	13	**Island**
19	8.8569×10^{-18}	20	*Void*
20	1.5533×10^{-18}	24	*Void*
21	3.7276×10^{-18}	22	*Void*
22	1.1345	11	
23	1.8450	10	
24	4.6690	7	

1	2	3	4	5	6	7	8
9	10	11	12	13	14	15	16
17	18	19	20	21	22	23	24
9	10	11	12	13	14	15	16
1	2	3	4	5	6	7	8

Fig. 7 Cell element numbering for strain energy filtering

vealed that the optimization method performs better when the mass constraint is gradually tightened with increasing refinement. The first reason is that a tight mass constraint produces slender, fine structures, which are difficult to represent by coarse cells. Therefore, starting the optimization with a tight mass constraint from the beginning is not a good strategy. Second, massive structures generated by a relaxed mass constraint are relatively simple to optimize initially, because fewer design degrees of freedom exist.

4 Numerical examples

4.1 Short cantilever optimized by VCL-GA

Figure 8 shows the results obtained by the VCL-GA for the case of a short cantilever beam. The design evolution starts with a domain refinement of 4×5 and gradually refines each cell in one or two dimensions (bisection). The final domain is discretized into 32×20 cells. The maximum allowable mass (2), is designated as M_0 in Fig. 8. Only half of the domain is modelled and optimized since symmetry about the horizontal axis is enforced. A population size of 50 is used for the first three stages, while the population size of the last stage is 150. The mutation rate is 1% and the crossover probability is set to 100%. Because good solutions are often destroyed by mating with bad designs, elitism is used where the best individual replaces the worst 30% of individuals in each generation. The mass constraint gradually decreases from 90% mass of the total design domain to 25%. IMOS (Milman et al. 1995) is used for the finite element analysis. It is well known that checkerboards can be restricted to some extent if quadratic finite elements are used. Because IMOS does not have quadratic elements, four linear elements are used to represent one design cell.

The first stage starts with a randomly generated population. The best solution of each stage is then transferred to the following generations by means of the diversity-generating scheme in (3). From the second generation on, the initial population designs are similar, because they are generated based on the best solution from the previous stage. Because the allowable mass decreases gradually, the structure becomes more slender with each refinement stage.

Figure 9 shows the chromosome length change for each stage. The length in the first stage is 20, and it increases to 640 in the last stage (more information is required to describe the more refined designs). The history of the objective function, compliance, is shown in Fig. 10. The first stage and second stage have two sub-stages each, and the third stage and fourth stage have three sub-stages each. The compliance shows a monotonous decrease in most sub-stages, but there are a few sub-stages where compliance does not converge monotonically. The overall compliance increases as the multi-stage search progresses. This is a consequence of the gradual imposition of the mass constraint.

Another interesting aspect is that a non-dominated front naturally emerges from the solutions obtained during the whole analysis. The approximate Pareto front explores the trade-off between mass and compliance, as shown in Fig. 11. Tightening the mass constraint does not increase the compliance severely during the early stages. However, as the total mass approaches 20%, the mass reduction is achieved by sacrificing considerable strength. This is an indication that most of the structure is used very efficiently at later stages of refinement.

4.2 Short cantilever optimized by a brute-force approach

It may be argued that the same 'optimal' solution can be obtained by an ordinary GA with the finest design domain resolution and that the seemingly complicated VCL method does not provide a measurable benefit. Figure 12 shows the solution obtained by the brute-force approach with approximately the same computational effort. The same strain energy filter is used to mitigate against protrusions and islands. The mass constraint is 25%, which is the same value as that imposed in the last stage of Fig. 8. The filter does remove some islands and protrusions, but there remain residual islands and protrusions because the original solution before filtering is very noisy. Some checkerboard patterns are also observed. It appears that, while the solution obtained by this brute-force approach does not have a clear structure,

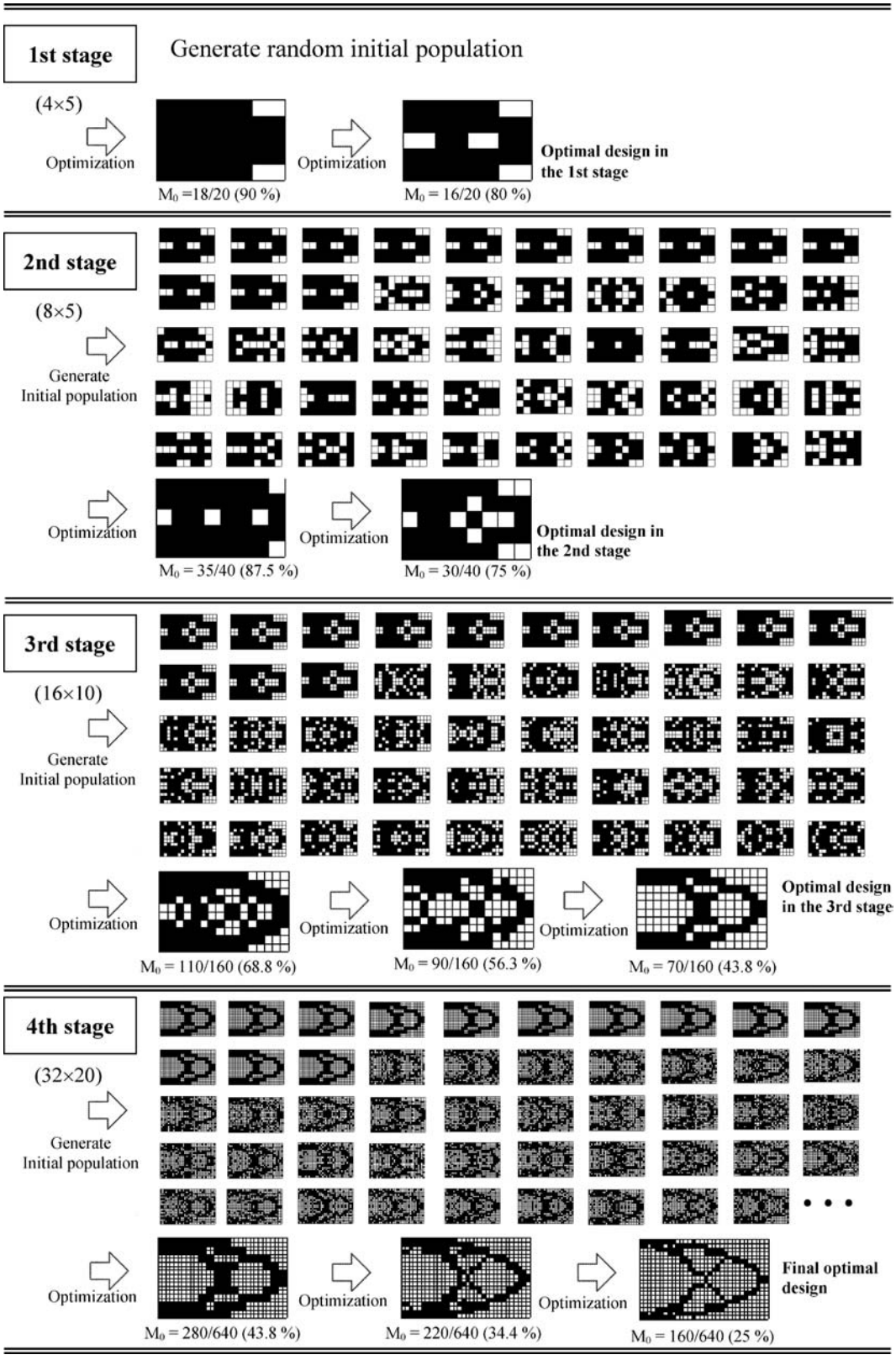


Fig. 8 Full design evolution for short cantilever design optimization with VCL-GA

the variable chromosome length GA produces a cleaner solution with the familiar X-shaped supports being revealed in the structure. Figure 12 compares the performance of the brute-force method and the variable chromosome length GA. With similar numbers of function evaluations, the objective function value (compliance) obtained by the variable

chromosome length GA is lower than that resulting from a single-stage search.

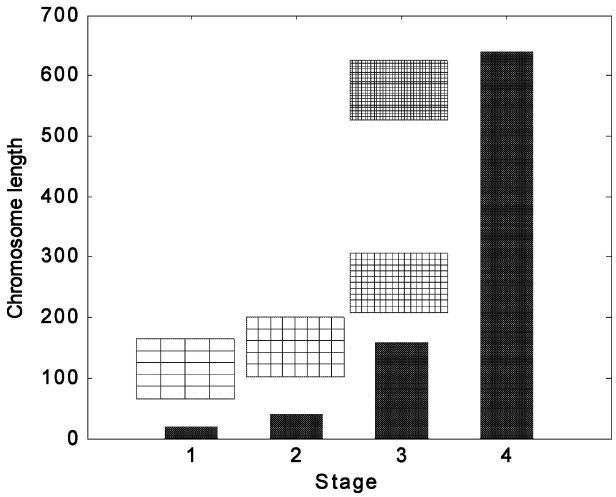


Fig. 9 Chromosome length change for the short cantilever problem

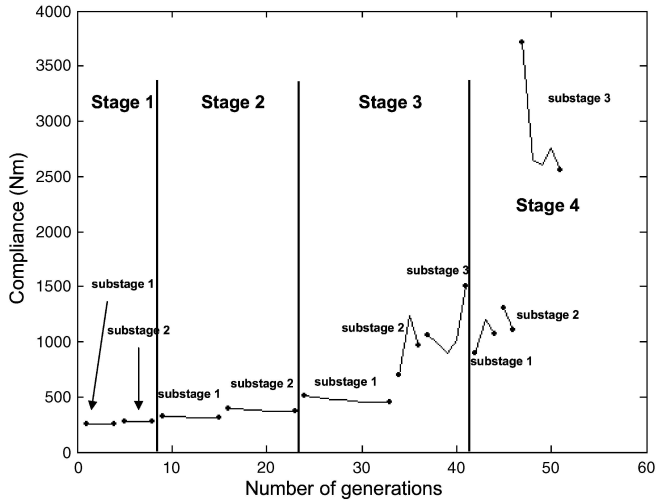


Fig. 10 Compliance history for the short cantilever problem

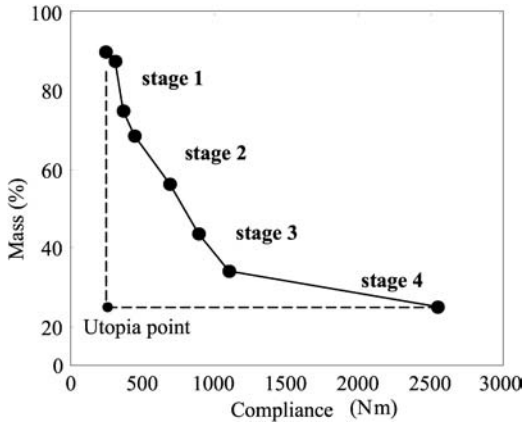


Fig. 11 Non-dominated front with mass plotted against compliance as objectives for the cantilever problem

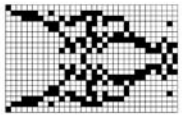
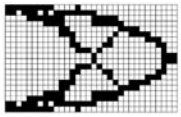
	Brute-force GA	VCL GA
Domain refinement resolution	32×20	32×20 (final resolution)
# of function evaluations	3386	3157
Compliance (N-m)	4780	2500
Result	 islands & protrusions checkerboards	 clearly X-shaped structure

Fig. 12 Performance comparison of the brute-force method and the variable chromosome length GA

The VCL-GA multi-stage search method has two advantages over the brute-force approach. First, the brute-force approach may not find a design as good as the variable chromosome length method can, because the design space for exploration is extremely large and it is very difficult to explore the entire design space with reasonable population sizes. Second, the variable chromosome length method determines rough optimum designs in the early stages when computational cost is inexpensive, and these designs are used as starting points in the following stages. Therefore, even when the one-time approach can find a fairly good solution, its computational cost is likely to be more expensive than that of the variable length approach.

4.3 Short cantilever sensitivity analysis

The previously found best solution (Fig. 8, stage 4) is obtained for a specific mass reduction history and population size. In this section, three different settings are investigated, as shown in Fig. 13. The final mass constraint is 25%, which is the same for all three cases. Case 2 has a different mass reduction history compared to Case 1 for the last three substages. Cases 1 and 2 have the same design evolution until the third stage, and in the fourth stage, they start to evolve into different solutions. Case 2 and Case 3 have the same mass constraint throughout the whole analysis, but the population size in the third stage is different. This leads to somewhat different solutions. The final solution is therefore dependent on the parameters chosen and the evolution path of the variable chromosome length GA. This might be seen as a disadvantage, but it also enables one to find several qualitatively different topologies during the early conceptual stage of design.

4.4 Bridge problem

In Fig. 14 a bridge problem is investigated using the VCL-GA method. A rigid roadway is given at the bottom of the design domain with a normalized density of unity. A uniformly distributed load is applied downwards underneath the domain, and both ends of the rigid domain are clamped. The

	Case 1	Case 2	Case 3
Stage	1	1	1
Mass constraint	18/20 (90%)	18/20 (90%)	18/20 (90%)
Population size	50	50	50
1	1	1	1
Mass constraint	16/20 (80%)	16/20 (80%)	16/20 (80%)
Population size	50	50	50
2	2	2	2
Mass constraint	30/40 (75%)	30/40 (75%)	30/40 (75%)
Population size	50	50	50
3	3	3	3
Mass constraint/	110/160 (69%)	110/160 (69%)	110/160 (69%)
Population size	50	50	100
3	3	3	3
Mass constraint	90/160 (56%)	90/160 (56%)	90/160 (56%)
Population size	50	50	100
3	3	3	3
Mass constraint	70/160 (44%)	70/160 (44%)	70/160 (44%)
Population size	50	50	100
4	3	3	3
Mass constraint/	260/640 (41%)	50/160 (31%)	50/160 (31%)
Population size	150	50	100
4	4	4	4
Mass constraint	220/640 (35%)	280/640 (44%)	280/640 (44%)
Population size	150	150	150
4	4	4	4
Mass constraint	160/640 (25%)	160/640 (25%)	160/640 (25%)
Population size	150	150	150
Compliance (N-m)	2420	2690	1940

Fig. 13 Different solutions for various settings of parameters

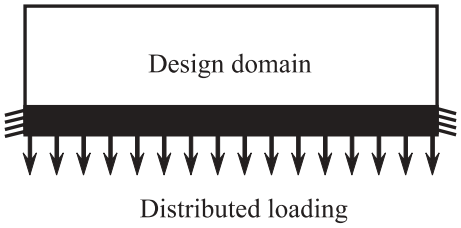


Fig. 14 Bridge under uniform loading

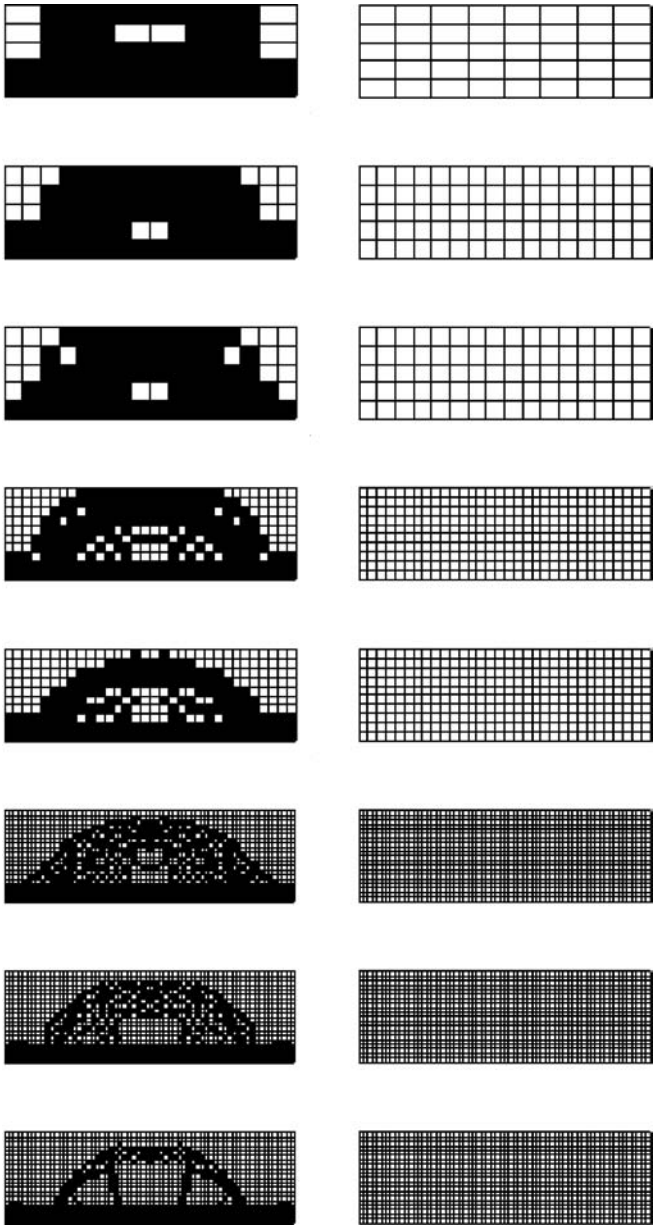


Fig. 15 Design optimization of bridge with VCL-GA

domain discretization starts with 8×4 and is subsequently refined to 16×4, 32×8, and 64×16. Symmetry is enforced about the vertical axis. As in the previous example, 30% elitism and a 1% mutation rate are used.

The solution history is shown in Fig. 15. The overall layout is found quickly in the first stage, and in the following

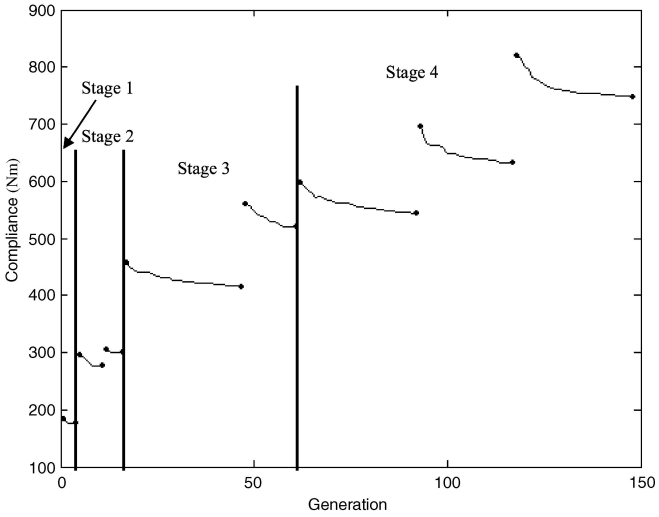


Fig. 16 Compliance history of bridge optimization with VCL-GA

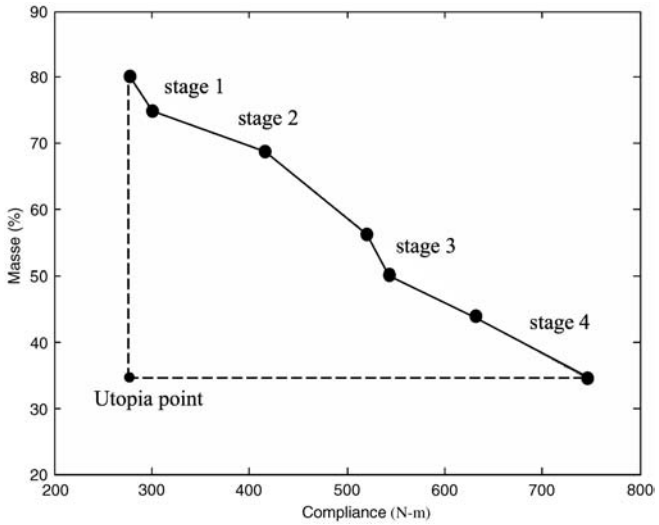


Fig. 17 Non-dominated trade-off between mass and compliance for bridge optimization with VCL-GA

stages, the solution becomes progressively more refined. In this example, some checkerboards are observed, but finally an arch bridge is obtained. It is observed that the ends of the arch do not meet with the ends of the rigid roadway. This is because the rigid roadway is quite thick, and only the middle region of the bridge needs additional support by the arch. The compliance converges in each sub-stage (Fig. 16), but the overall compliance increases as the stages progress due to the gradual imposition of the mass constraint. This gradual change in mass naturally generates a non-dominated front that shows the trade-off between mass and compliance, as shown in Fig. 17.

5 Conclusions and future work

A novel genetic algorithm (GA) in which the length of chromosomes evolves was developed for structural topology op-

timization. We call this method the variable chromosome length genetic algorithm, or VCL-GA for short. The design domain is refined gradually in a multi-stage search process. The proposed method is applied to two structural optimization problems: a short cantilever and a bridge problem. The algorithm increases the likelihood of approaching higher levels of fitness, allows the solution of problems with large numbers of design variables, and broadens the notion of design evolution to include gradually increasing levels of complexity. The solution is compared with the one-time brute-force approach, and it is demonstrated that the variable chromosome length GA finds a better solution with less computational cost.

The strain energy filter can be used when the objective function is compliance or displacement. If eigenvalue problems or thermal problems are considered, a new metric for filtering that identifies inefficiently used elements for the particular discipline will be needed. As discussed in Sect. 4.3, the final optimal solution obtained depends on several parameters: the number of stages, the mass constraint imposition history, and the population size for each stage. Of course, the usual GA parameters, such as selection level, crossover rate, mutation rate, and maximum number of generations or termination criteria, affect the solution as well. Because we use the bisection method for refinement, which divides each cell into two or four cells, the number of stages is naturally determined for a given first and final resolution. The choice of number of sub-stages, implemented for the mass constraint tightening, in each stage is up to the user. We found that mass constraint schedules with rapid tightening often produce inferior local solutions.

In this work, there is only a mass constraint which is imposed gradually resulting in non-dominated solutions. The mass constraint will play the same role even when there are other constraints. Mass is closely related to the number of solid (or void) cells in a stage, and large mass (= a small number of void cells) in the initial stages reduces the number of design choices and the design space becomes relatively simple. This helps quickly finding good solutions that can be used as seeds for subsequent stages. It would be interesting to compare the results obtained here with true multi-objective optimization using genetic algorithms (MOGA). This is left as future work.

An adaptive chromosome length GA, in which chromosome length changes adaptively according to the characteristics of a problem, will be developed as future work. This will also allow chromosomes to contract once the design has converged according to a certain schema. The relative efficiency of optimization methods can be assessed using information theory (Krus and Andersson 2003). Several convergence criteria for outer-loop convergence will also be developed and tested. We acknowledge the fact that for some problems genetic algorithms may not be the best solution and that sensitivity-based methods might be superior in some cases. Nevertheless, the results presented here demonstrate the promise of adaptive, variable chromosome length GAs, particularly when implemented in a parallel computing environment.

References

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

Bendsøe MP (1995) Optimization of structural topology, shape, and material. Springer, Berlin Heidelberg New York

Cea J (1981) Numerical methods in shape optimal design. Optimization of distributed parameter structures. Sijthoff & Noordhof, Netherlands, pp.1049–1087

Chapman CD, Saitou K, Jakiela MJ (1994) Genetic Algorithms as an approach to configuration and topology design. J Mech Des Trans ASME 116:1005–1012

Chapman CD, Jakiela MJ (1996) Genetic algorithm-based structural topology design with compliance and topology simplification considerations. J Mech Des Trans ASME 118:89–98

DeRose GCA, Diaz AR (2000) Solving three-dimensional layout optimization problems using fixed scale wavelets. Comput Mech 25:274–285

Diaz AR, Bendsøe MP (1992) Shape optimization of structures for multiple loading conditions using homogenization method. Struct Optim 4:17–22

Diaz AR (1999) A wavelet-Galerkin scheme for analysis of large-scale problems on simple domains. Int J Numer Methods Eng 44:1599–1616

Eschenauer HA, Olhoff N (2001) Topology optimization of continuum structures: a review. Appl Mech Rev 54:331–389

Goldberg D (1989) Genetic algorithms in search, optimization, and machine learning. Addison-Wesley, Reading, MA

Guan H. et al. (2003) Bridge topology optimization with stress, displacement and frequency constraints. Comput Struct 81:131–145

Haftka RT, Grandhi RV (1986) Structural shape optimization — a survey. Comput Methods Appl Mech Eng 57:91–106

Hajela P, Lee E (1995) Genetic algorithms in truss topological optimization. J Solids Struct 32:3341–3357

Haug EJ, Choi KK, Komkov V (1986) Design sensitivity analysis of structural systems. Academic, Amsterdam

Holland J (1975) Adaptation in natural and artificial systems. University of Michigan Press, Ann Arbor, Michigan

Jakiela MJ et al. (2000) Continuum structural topology design with genetic algorithms. Comput Methods Appl Mech Eng 186:339–356

Kim H, Querin OM, Steven GP (2002) On the development of structural optimisation and its relevance in engineering design. Des Stud 23:85–102

Kim IY, Kwak BM (2002) Design space optimization using a numerical design continuation method. Int J Numer Methods Eng 53:1979–2002

Kim JE, Jang GW, Kim YY (2003) Adaptive multiscale wavelet-Galerkin analysis for plane elasticity problems and its applications to multiscale topology design optimization. Int J Solids Struct 40:6473–6496

Kim YY, Yoon GH (2000) Multi-resolution multi-scale topology optimization — a new paradigm. Int J Solids Struct 37:5529–5559

Kirsch U (1989) Optimal topologies of structures. Appl Mech Rev 42:223–239

Krus P, Andersson J (2003) Optimizing optimization for design optimization. Proceedings of DETC'03, ASME 2003 Design Engineering Technical Conferences and Computers and Information in Engineering Conference, Chicago, Illinois, USA, September 2–6

Kwak BM (1994) A review on shape optimal design and sensitivity analysis. JSCE J Struct Eng/Earthquake Eng 10(4):159s–174s

Lin C-Y, Hajela P (1994) Design optimization with advanced genetic search strategies. J Adv Eng Softw 21:179–189

Lin C-Y, Hajela P (1993) Genetic search strategies in large scale optimization. Proceedings of the 34th AIAA/ ASME/ASCE/ AHS/ASC SDM Conference, La Jolla, California, 2437–2447

Mackerle J (2003) Topology and shape optimization of structures

using FEM and BEM — a bibliography (1999–2001). Finite Elem Anal Des 39:243–253

Maute K, Ramm E (1995) Adaptive topology optimization. Struct Optim 10:100–112

Michell AGM (1904) The limits of economy of material in frame-structures. Philos Mag 8:589–597

Milman MH, Briggs HC, Ledeboer W, Molody JW, Norton RL, Needels L (1995) Integrated modeling of optical systems user's manual, release 2.0. JPL D-13040

Min S, Nishiwaki S, Kikuchi N (2000) Unified topology design of static and vibrating structures using multiobjective optimization. Comput Struct 75:93–116

Rousselet B (1981) Implementation of some methods of shape design. Optimization of distributed parameter structures. Sijthoff & Noordhof, Netherlands

Rozvany GIN, Bendsøe MP, Kirsch U (1995) Layout optimization of structures. Appl Mech Rev 42:41–119

Ryoo J, Hajela P (2004) Handling variable string lengths in ga based structural topology optimization. Struct Multidisc Optim 26:318–325

Todoroki A, Haftka RT (1997) Stacking sequence matching by a two-stage genetic algorithm with consanguineous initial population. Proc. of the 1997 38th AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics and Materials Conf., Part 2 (of 4), Paper AIAA-97-1228, 1297–1302

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

Zolésio JP (1981) The material derivative (or speed) method for shape optimization. Optimization of distributed parameter structures. Sijthoff & Noordhof, Netherlands, pp.1089–1151