Evolution of Contours for Topology Optimization

Gideon Avigad, Erella Matalon Eisenstadt, Shaul Salomon,
and Frederico Gadelha Guimar

Abstract. Topology optimization is used to find a preliminary structural configuration that meets a predefined criterion. It involves optimizing both the external boundary and the distribution of the internal material within a structure. Usually, counters are used a posteriori to the topology optimization to further adapt the shape of the topology according to manufacturing needs. Here we suggest optimizing topologies by evolving counters. We consider both outer and inner counters to allow for holes in the structure. Due to the difficulty of defining a reliable measure for the differences among shapes, little research attention has been focused on simultaneously finding diverse sets of optimal topologies. Here, niching is implemented within a suggested evolutionary algorithm in order to find diverse topologies. The niching is then embedded within the algorithm through the use of our recently introduced partitioning algorithm. For this algorithm to be used, the topologies are represented as functions. Two examples are given to demonstrate the approach. These examples show that the algorithm evolves a set of diverse optimal topologies.

1 Background

The background for the methodology of this paper includes genetic algorithms (see Section 1.1), which serve here as the search algorithm, and their use for optimizing topologies (see Section 1.2). Because we aim at finding multiple solutions to a single objective problem, multi-modal optimization through the utilization of niching is of interest. Therefore, a review of niching approaches is given in Section 1.3. Finally, our recently suggested approach to find diverse sets of functions is briefly discussed

Gideon Avigad · Erella Matalon Eisenstadt · Shaul Salomon
Department of Mechanical Engineering, Ort Braude College, Karmiel, Israel
e-mail: {gideona,erella,shaulsal}@braude.ac.il

Frederico Gadelha Guimar
Department of Electrical Engineering, Federal University of Minas Gerais, Brazil
e-mail: fredericoguimaraes@ufmg.br

O. Schütze et al. (Eds.): EVOLVE – A Bridge between Probability, AISC 175, pp. 397–412.
springerlink.com

in Section 1.4. In the paper, this approach will serve to enhance the search towards diverse topologies.

1.1 Evolutionary Algorithms

Evolutionary algorithms (EAs) belong to a class of non-gradient methods that have grown in popularity following the original publication of [20], and later [8]. expanded the idea and helped make it popular. EAs are stochastic search methods that mimic natural biological evolution. EAs operate on populations of potential solutions, applying the principle of survival of the fittest to produce better and better solutions. An EA uses a population of individuals (solutions) instead of a single solution to perform a parallel search in the problem space. At each generation, a new set of approximations is created by a nature-inspired process. The natural processes commonly mimicked by EAs include selection, breeding, mutation, migration, and survival of the fittest.

1.2 EC for Size/Shape/Topology Optimization

According to [15], the problems addressed by structural optimization can be divided into three major categories, namely: a. Topology (layout) Optimization where the search is for an optimal material layout; b. Shape Optimization, where the search targets the optimal contour, or shape, of a structural system whose topology is fixed; and c. Sizing Optimization, in which the optimization aims at optimal cross-sections, or dimensions, of elements of a structural system whose topology and shape are fixed. Finding a good structural configuration (topology) prior to shape and sizing optimization is an important but difficult task. In comparison to shape optimization, topology optimization is more complex since it involves the optimization of both the external boundary and the distribution of the internal material within a structure. Topology optimization is used to find a preliminary structural configuration that meets a predefined criterion. Occasionally it yields a design that can be completely new and innovative. An EC approach to the continuum topology optimization design problem based on genetic algorithms has been proposed, e.g., in [21] and in [22]. In these studies, the design domain was discretized into small elements containing materials or voids in a cantilever plate so that the structure's weight was minimized subject to displacement and/or stress constraints. A variety of structural design fitness functions, among them stiffness, area, and perimeter, were employed to find optimal cantilevered plate topologies. Another coding approach for topologies is based on graph theory [23], in which a topology is represented by a connected simple graph consisting of vertices and simple undirected cubic Bezier curves with varying thicknesses. The derived results show that the graph representation EA can generate clearly defined and distinct geometries and perform a global search. A bit-array representation EA [24] was also implemented for topology optimization. Design connectivity and constraint handling were further developed to improve the efficiency of the EA. In addition, a violation penalty method has been proposed to drive the EA search towards those topologies with higher structural performance, less unusable material and fewer separate objects in the design domain. A multi-EA system [25] and a variable chromosome length genetic algorithm [16] were proposed for continuum structure topological optimization. Recently, a two-stage adaptive genetic algorithm (TSAGA) [4] was developed in bit-array represented topology optimization. The authors demonstrate the efficiency and effectiveness of TSAGA in comparison to other approaches in reaching global optimal solutions on several case problems.

1.3 Niching within EC and for Topology Optimization

The sharing method, originally suggested by Holland, [8], is probably the most popular niching technique. Sharing is analogous to a situation in nature in which the resources of a niche have to be shared. In mathematical terms this method penalizes solutions that are similar by dividing the fitness of the niche among them. According to [19], niching methods can be divided into iterative methods, explicit parallel sub-population methods and implicit parallel sub-population methods. Iterative methods address the problem of locating multiple optima of a multi-modal function by repeatedly applying the same optimization algorithm. Several techniques have been used to avoid iterations towards local minima, such as the Tabu technique [7], the sequential niche technique [4] and various jump techniques [12]. Explicit parallel sub-population methods attempt to generate multiple solutions to a multi-modal optimization problem by dividing a population into sub-populations that evolve in parallel. These methods include Multiple-National EA [25], Island EAs [5], the Adaptive Isolation Model [5], and Particle Swarm Optimization [18]. Without communications among the populations, these methods are similar to iterative methods. Implicit parallel sub-population methods attempt to generate multiple solutions by introducing niche/speciation techniques so that population diversity is maintained and many niches survive in a single population. Among these methods are crowding [22], [20], fitness sharing [8], restricted tournament selection [13], species conservation techniques [20] and Genetic Sampler [20]. Species conservation is a relatively new technique for solving multi-modal optimization problems and has been proved effective for obtaining multiple solutions of tested multi-modal problems (e.g., [22]). The study reported in [10] proposed a new approach for finding diverse solutions to a multi-modal problem. The authors suggested posing the single multi-modal problem as a bi-objective problem, with the value of the function as the first objective and the number of neighboring points that are better than others as the second objective. It should be noted that the above studies deal with single objective problems. Although the notion of resource sharing is also an essential part of evolutionary multi-objective optimization this aspect is beyond the scope of the current paper. Niching within topology optimization has received very little research attention. The few existing studies include the search for optimal topologies for trusses and space planners (see e.g., [2], [17]). There is no record of a simultaneous search for several optimal beam cross sections (topologies) using niching within evolutionary computation.

1.4 Function Diversity

The partitioning of the set of functions into subsets (Avigad et al., 2012) is explained in the following (taken from [1]). This partitioning serves as the basis for the approach taken in this paper.

Let $A = \{f_c(t)\}_{(c\in\Omega, t\in D)}$, be a set of real valued alternative functions, each sampled k times, with $c = [c_1, \ldots, c_p]^T$, $\Omega \in \mathbb{R}^p$, $D \in \mathbb{R}$ and $f_c(t) = [f_c(t_1), \ldots, f_c(t_k)]^T$.

$$a_j = \min_{c\in\Omega} f_c(t_j), \quad b_j = \max_{c\in\Omega} f_c(t_j), \quad \Delta_j = b_j - a_j, \;\; j = 1, \ldots, k, \quad dA = \max_{j=1,\ldots,k} \Delta_j$$

In fact, dA is the diameter of the set A in the Chebyshev metric. The idea of the suggested algorithm is that at each step, the set with the largest diameter is partitioned into two subsets. The result of such a partitioning is that subsets are formed, each with a smaller partition than the former maximal partition. This process is repeated until the desired number of subsets is attained. The following algorithm describes the suggested procedure.

Partitioning the set A to m subsets

(a) $A_1 \leftarrow A$
(b) $i \leftarrow 1$
(c) $S \leftarrow \{A_1, \ldots, A_i\}$
(d) **While** $i < m$ **do**
(e) Find A_j such that $dA_j = \max_{l=1,\ldots,k} dA_l$
(f) Find sample t_p such that $\Delta_p = dA_j$. assemble subsets A_{j1}, A_{j2} from functions $f_c(t) = [f_c(t_1), \ldots, f_c(t_k)]^T$ which satisfy the inequalities $f(t_p) \leq a_p + dA_j/2$ and $f(t_p) > a_p + dA_j/2$ respectively
(g) $S \leftarrow \{A_1, \ldots, A_{j1}, A_{j2}, \ldots, A_i\}$
(h) $i \leftarrow i + 1$
(i) **End while**

To demonstrate the approach, [1] used an artificial set of 100 functions and divided it into six different sub-sets by using the partitioning algorithm. Figure 1 (borrowed from [1]) shows the results of the first, second, third and fifth partitioning stages (resulting in six subdivisions of the function space). As a last step one function is chosen from each sub-division. In [1], such functions were evolved by using set-based

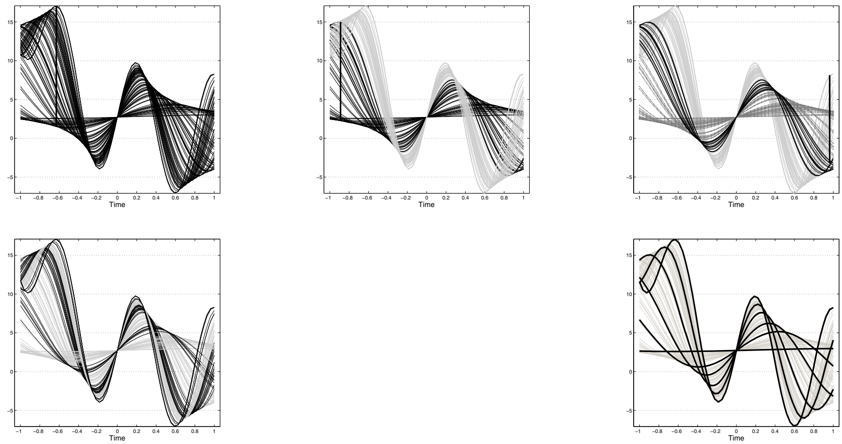


Fig. 1 Partitioning algorithm

dominance and by implementing crowding, which was based on the number of functions present in each sub-division. The current study proposes evolving contours to optimize topologies by utilizing both the partitioning algorithm and crowding, as suggested in [1].

Evolving contours for the purpose of evolving topologies differs from the claim (see Figure 1) that contours are used only for shape optimization. Moreover, the purpose here is not merely to find one optimal topology but rather to look for several diverse optimal topologies. The motivation for finding different solutions to an optimization problem is well known (see e.g., Mattson and Messac, 2005). Basically, finding different solutions provides decision makers with greater flexibility in choosing the preferred design based on unmodeled properties.

2 Methodology

To enhance topology diversity, the algorithm proposed in [1]is adopted and adapted in this paper. The algorithm in [1], deals with functions. This means that in order to use that algorithm for topologies, the topologies should also be described by functions.

2.1 Describing Topologies by Functions

In this paper, 2D topologies are considered. Polar coordinates are used here for describing topologies, although, as discussed, other coordinate systems may be used. Topologies with and without holes are considered.

For topologies without a hole, the j-th contour is described by an ordered set of radii: $r_j = [r_1^j(\theta_1), \ldots, r_{N_\theta}^j(\theta_{N_\theta})]^T$ where $\theta \in D \subset R$ is an equally spaced angle. The topology is formed by connecting the ordered set to form a continuous function (through linear sections). The left panel of Figure 2 depicts the encoding of a topology with k points, and the middle panel of that figure depicts the formed topology. It should be noted that in this paper we use reflection symmetry (or mirror symmetry) with respect to the horizontal and vertical axes, although the use of other symmetries or of no symmetry at all may be considered. Reflection symmetry implies that a quarter of the contour may describe the entire contour.

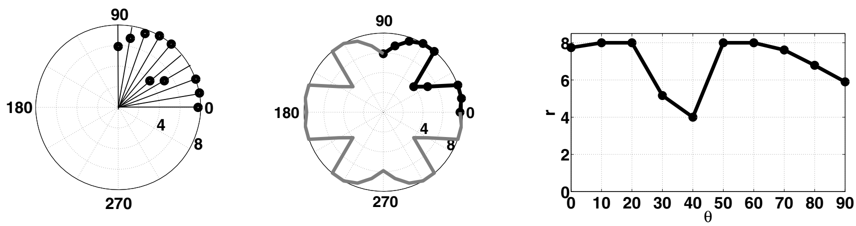


Fig. 2 The encoding of topology without a hole

In fact, based on these equally spaced coordinates we may now describe the topology as a function of θ, so that the j-th topology may be described as a function $f_r(\theta) = r(\theta)$. The middle panel of Figure 2 depicts the topology while the right panel depicts its plot as a function.

Two contours are used to describe a topology with a hole. The j-th topology is coded by its inner contour radius $R_j^{in} = [r_1^{j,in}(\theta_1), \ldots, r_{N_\theta}^{j,in}(\theta_{N_\theta})]^T$ and by its outer contour radius $R_j^{out} = [r_1^{j,out}(\theta_1), \ldots, r_{N_\theta}^{j,out}(\theta_{N_\theta})]^T$. The topology is formed by connecting the ordered set to a continuous function (linear sections). The left panel of Figure 3 depicts the code of the topology, where squares and circles designate the inner and outer radii of the inner and outer contours respectively. The middle panel of the figure shows the topology formed by connecting the radii by linear sections and using the rotational reflection.

As in the no-hole case, here also this topology may be described by a function so that the j-th topology may be described as functions; $f_{r_{in}}^{r_{out}}(\theta) = \{f^{r_{out}}(\theta) \cup f_{r_{in}}(\theta)\}$. It should be noted that the outer and inner boundaries of the outer and inner contours should be determined to ensure a reasonable thickness for the topology. The middle panel of Figure 3 depicts the topology, and the right panel shows its description as a function.

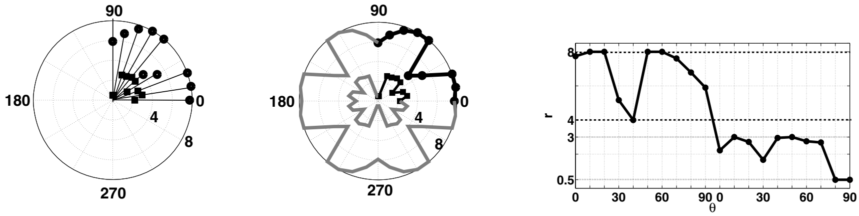


Fig. 3 The encoding of topology with hole

2.2 Problem Definition

The optimization problem is defined as follows: For topologies without a hole:

$$\begin{aligned} &\min_{r} F(f_r(\theta)) \\ &\forall r \in R : r_{min} \leq r \leq r_{max} \\ &g(f_r(\theta)) \geq 0 \end{aligned} \tag{1}$$

For topologies with a hole:

$$\begin{aligned} &\min_{r^{out},r^{in}} F(f_{r_{in}}^{r_{out}}(\theta)) \\ &\forall r^{in} \in R^{in} : r^{in}_{min} \leq r^{in} \leq r^{in}_{max} \\ &\forall r^{out} \in R^{out} : r^{in}_{max} + \delta \leq r^{out} \leq r^{out}_{max} \\ &g(f_{r_{in}}^{r_{out}}(\theta)) \geq 0 \end{aligned} \tag{2}$$

where $F : F(f(\cdot)) \rightarrow 0 \in \Gamma \subset R$ and Γ is the objective space. For example $F(f(\cdot))$ can be a cross section area, the first or the second moment of inertia and other topology dependent characteristics.

2.3 Diverse Topologies and Niching

Niching is implemented to find a diverse set of topologies. For this purpose, a partitioning algorithm is used. First the following parameters must be computed: $a_j = min\ f(\cdot)$, $b_j = max\ f(\cdot)$ and $\Delta_j = b_j - a_j, j = 1,\ldots,k$ as well as $dA = \max_{j=1,\ldots,k} \Delta_j$ then the algorithm of [1] is used by slight changes as follows:

Partitioning the set A to m subsets

(a) $A_1 \leftarrow A$
(b) $i \leftarrow 1$

(c) $S \leftarrow \{A_1, \ldots, A_i\}$
(d) **While** $i < m$ **do**
(e) Find A_j such that $dA_j = \max\limits_{l=1,\ldots,k} dA_l$
(f) Find sample θ such that $\Delta_p = dA_j$. Assemble subset A_{j1}, A_{j2} from functions $f_r(\theta)$ or $f_{r_{in}}^{r_{out}}(\theta)$ which satisfy the inequalities:
$f_r(\theta) \leq a_p + dA_j/2$ or $f_{r_{in}}^{r_{out}}(\theta) \leq a_p + dA_j/2$ and
$f_r(\theta) > a_p + dA_j/2$ or $f_{r_{in}}^{r_{out}}(\theta) > a_p + dA_j/2$ respectively
(g) $S \leftarrow \{A_1, \ldots, A_{j1}, A_{j2}, \ldots, A_i\}$
(h) $i \leftarrow i + 1$
(i) **End while**

The algorithm partitions a population of topologies (represented as functions) to N predefined sub-sets. A solution will reproduce according to its fitness (value of $F(f(\cdot))$) which will be penalized according to the number of functions with which it shares the same sub-partitioning (same niche). Such a penalty will prevent genetic drift and will increase the chances that other optimal solutions will reproduce. For example, Figure 4 depicts 20 topologies represented by their related

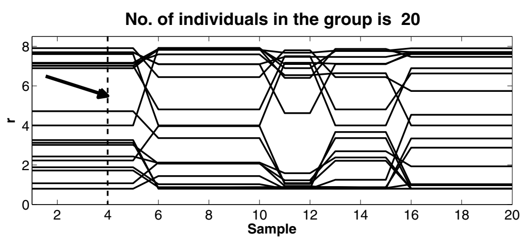


Fig. 4 20 individuals before division

functions. The dashed line marks the sample with the largest difference and therefore the point where the algorithm will divide the group into two groups. This partitioning results in 5 and 15 individuals in each subdivision, as depicted in Figure 5. The next partition is marked by the dashed line in the right panel of Figure 5. The result of dividing the second group (of 15 individuals) into two

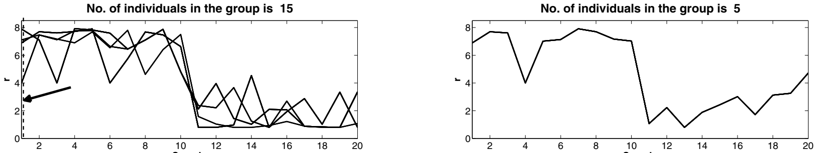


Fig. 5 The two formed subdivisions containing 5 (left panel) and 15 (right panel) individuals

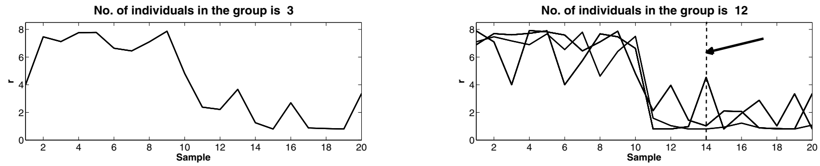


Fig. 6 The second and third groups

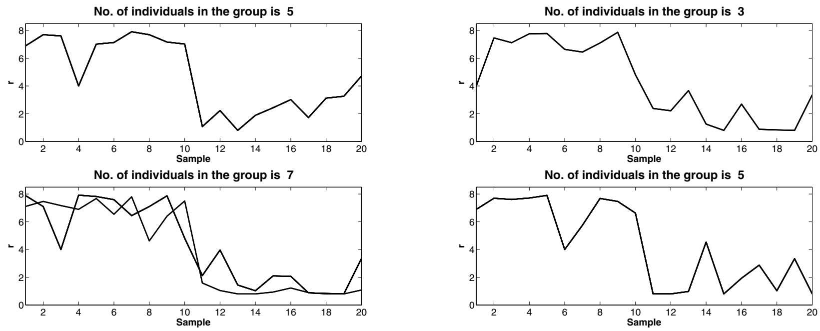


Fig. 7 The second and third groups

groups is depicted in Figure 6. The point where the algorithm will divide again to form the next two groups is marked by a dashed line. Finally, the four groups are presented in Figure 7.

2.4 The Evolutionary Optimization Algorithm

The algorithm described here aims at searching for optimal topologies based on the optimization problem set in Section 2.2. The algorithm utilizes an archive in order to preserve elite solutions.

The Evolutionary Optimization Pseudo Code

(a) Initialize a population P_t with n individuals. Also, create a population $Q_t = P_t$
(b) **While** $t \leq$ predefined number of generations
(c) Combine parent and offspring populations and create $R_t = P_t \cup Q_t$
(d) For each individual $z \in R_t$ compute:

 (d.1) $F(f(z))$
 (d.2) $g(f(z))$
 (d.3) The niche count: $m(z)$
 (d.4) The fitness: $fit(z) = F(f(z))\frac{2n}{2n-m(z)}$

(e) Create elite population P_{t+1} of size n from R_t (procedure I)
(f) Create offspring population Q^*_{t+1} of size n from R_t by Tournament selection (procedure II)
(g) Perform Crossover to obtain Q^{**}_{t+1} from Q^*_{t+1}.
(h) Perform Mutation to obtain Q_{t+1} from Q^{**}_{t+1}.
(i) Set $t = t+1$ and go to (b).
(j) Provide the decision maker with N topologies, each from a different subdivision and each possessing $min(fit)$ within this subdivision.

Procedure I: Archiving

Compute n_c = number of individuals that comply with $g(f(z)) > 0$

(a) If $n_c \leq n$ then include the n_c individuals in P_{t+1} and sort the remaining $2n - n_c$ individuals to a list $L1(g(f(z) :<))$. Add the first $n - n_c$ individuals of $L1$ to P_{t+1}.
(b) If $n_c > n$
   (b.1) Include N individuals in P_{t+1} taking one individual from each sub division that is with the min(over all fit).
   (b.2) Then list all $n_c - N$ individuals to a list $L2(fit(f(z) :>))$. Add the first $n - N$ individuals of $L2$ to P_{t+1}.

Procedure II: Tournament Selection

(a) If $g(f(z)) \geq 0 \wedge g(f(z')) < 0$ include z in Q_{t+1}
(b) If $g(f(z)) < 0 \wedge g(f(z')) < 0$
Then if $g(f(z)) > g(f(z'))$ include z in Q_{t+1}
(c) If $g(f(z)) \geq 0 \wedge g(f(z')) \geq 0$
Then if $fit(f(z))\ leqfit(f(z'))$ include z in Q_{t+1}

Algorithm explanation
In step (a), two populations are initialized. A "while loop" begins at step (b) and ends at step (i). In this loop populations of candidate solutions are evolved. Step (c) is a common step used in MOEAs, which involves creating a combined population. In step (d), the following calculations take place with respect to each individual within a population: (d.1)- value of the objective function (which is to be optimized); (d.2)- value of the constraint; (d.3)- number of functions that share the same subdivision with the individual; (d.4)- fitness value of the individual, which is computed by penalizing the objective function's value. A greater number of shared individuals results in a higher penalty and therefore decreased fitness. In step (e) the elite population is formed by utilizing Procedure I. The procedure admits feasible optimal (low fitness value; without loss of generality) solutions to the archive. Moreover, it ensures that the archive will always include at least one representative from each of the N subdivisions. In step (f) the offspring population is created by using Procedure II. In each tournament (among two arbitrary selected individuals), the winning individual is feasible and possesses a lower fitness value (without loss of generality)

than its competitor. If both competitors are infeasible, the one with a lesser violation of the constraint is the winner. Steps (g) and (h) are the crossover and mutation steps respectively. In the final step (step (j)), the decision maker is presented with N solutions that have the lowest fitness value (the best as far as minimization is considered). In order to preserve diversity, each of the presented topologies is extracted from another sub-division. Please note that step (j) is an a posteriori step and the decision maker is not involved within the evolutionary search.

3 Examples

In this section two examples are used to demonstrate the proposed approach and to highlight its potential for solving real life topology optimization problems.

3.1 Cross Section Optimization for a Structure Subjected to a Tensile Force

A beam is subjected to a tensile force of magnitude: $F_r = 150kN$. The objective of the optimization is to find a topology with a minimal cross section A. The allowed normal stress is: $[\sigma] = 10MPa$, therefore

$$A_{min} \geq \frac{F_r}{[\sigma]} = \frac{150\times10^3}{10\times10^6} = 150cm^2.$$

This implies on the constraint:

$$g = A - A_{min} = A - 150 \geq 0$$

The search here will be conducted towards topologies that contain a hole. The topology search space is a priori set such that:
$0.5 \leq r^{in} \leq 3; 3+1 = 4 \leq r^{out} \leq 8$, where dimensions are given in cm.

The initial population includes 100 individuals and is represented as functions in the left panel of Figure 8. In order to prevent the outer contour from dominating the diversity, normalization is applied such that the amplitude of both contours is the same. The normalized functions are depicted in the right panel of Figure 8. Note that the radii have not been altered and that the inner radius seems bigger due to the normalization.

Two topologies arbitrarily chosen from the initial population are depicted in Figure 9. These are clearly non-optimal solutions because the topology in the left panel involves a cross section that is much too big according to the constraint, whereas the topology in the right panel has a cross section that is smaller than allowed.

Apart from diversity preservation, the algorithm has evolved optimal topologies. Figures 10 and 11 depict the resulting optimal diversified topologies when the proposed algorithm is run with N=2 and N=4, respectively. Clearly the algorithm has

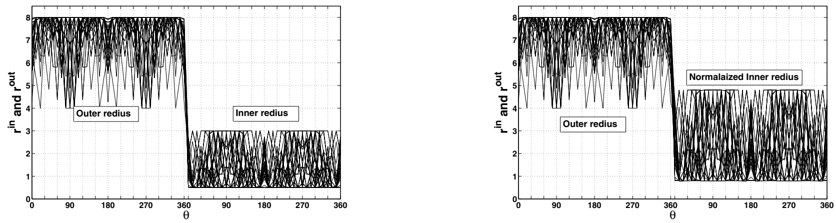


Fig. 8 Normalized and Un-normalized function

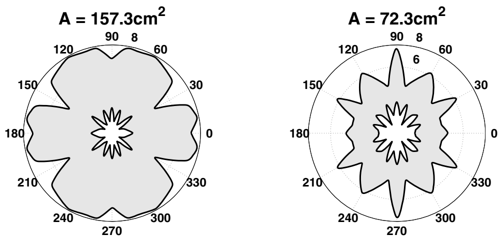


Fig. 9 Two cross section in the first generation

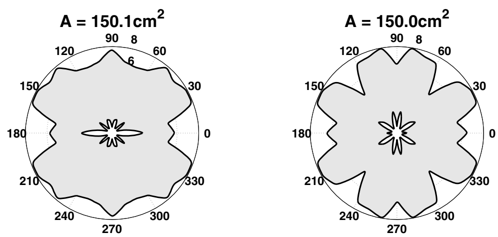


Fig. 10 The result for N=2

evolved diverse optimal topologies. All are distinctly different and possess the minimal (or close to the minimal) possible cross section area. Considering feasibility, clearly most topologies are not natural candidates for manufacturing. Nevertheless, as expected from such algorithms, they have shown the way, and adaptations to facilitate manufacturing should follow. One approach that should help is to encode the topologies with less complexity, i.e., fewer angle divisions. Figure 3.1 depicts the same problem; however, instead of using 10 angle divisions to decode the topologies, 5 divisions are used.

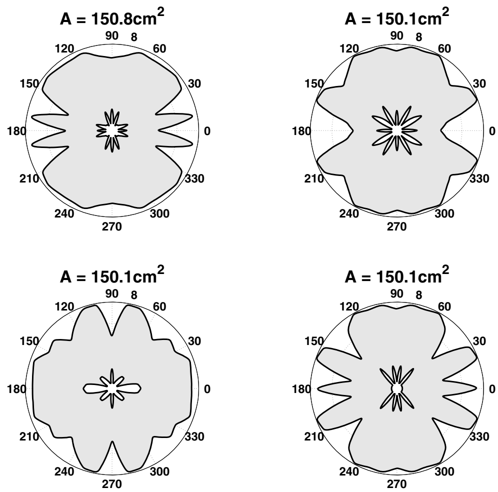
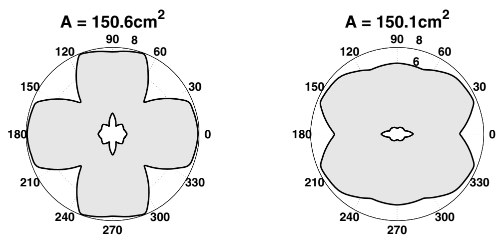


Fig. 11 The result for N=4



3.2 Cross Section Optimization for a Structure Subjected to a Moment

A beam is subjected to a pure moment. The moment applied magnitude is: $M = 66 \times 10^3 N \times m$. The allowed normal stress is: $[\sigma] = 120MPa$ The equation connecting the allowable stress and moment is: $[\sigma] \geq \frac{My_{max}}{I_{xx}} = \frac{M}{S_x}$ where I_{xx} is the second moment of inertia, y_{max} is the the maximal perpendicular distance from axis x, and $S_x = \frac{I_{xx}}{y_{max}}$ is the section modulus about x axis. The allowable stress determined the allowable range of section modulus about the x axis:

$$[\sigma] \geq \frac{M}{S_x} \rightarrow S_x \geq \frac{M}{[\sigma]} = \frac{66 \times 10^3}{120 \times 10^6} = 5.5 \times 10^{-4} m^3 = 550 cm^3$$

The constraint is $g = S_x - 550 \geq 0$ and the objective is the minimal beam weight. Because the problem deals only with constant cross section beams, minimizing the weight is the same as minimizing the beam's cross section area. Figure 12 depicts three evolved topologies with no holes. It is apparent that although different

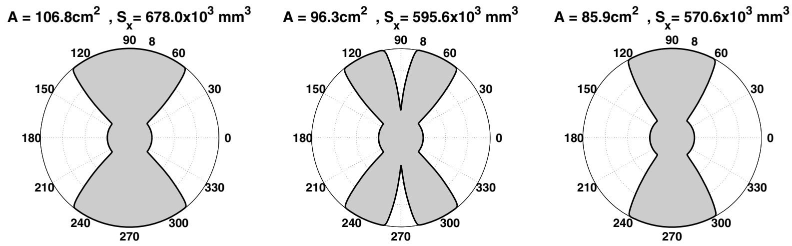


Fig. 12 The result for N=3 decoded without a hole

topologies have been evolved, their cross sections are not similar. As expected, the optimal topology involves an I-shaped cross section, where other topologies are less optimal versions of it.

4 Summary and Conclusions

In this paper, we have adapted a recently proposed partitioning algorithm in order to evolve a diverse set of optimal topologies. It has been suggested to code topologies using polar coordinates. Topologies with holes and without holes were coded. The coding was used to establish topologies through linear interpolation. Then, the topologies were represented as functions (of an equally spaced angles). Once this was achieved, the partitioning algorithm was implemented within an evolutionary search. This algorithm enhances a search toward a diverse set of optimal topologies. These diversified optimal topologies are associated with multiple optimal solutions (if the problem is inherently multi modal) or with different levels of optimality (if the problem is not multi modal by nature). In future work other coordinates will be used to describe the topologies (Cartesian), allowing more complicated structure with more holes. Moreover, 3D shapes should be evolved by describing the added dimension using a function as well. Real life problems should be considered, and comparisons with other approaches should be made.

Acknowledgements. This research was supported by a Marie Curie International Research Staff Exchange Scheme Fellowship within the 7th European Community Framework Programme.

References

1. Avigad, G., Goldvard, A., Salomon, S.: Partitioning Algorithms. Technical report No. br352012, `http://brd.braude.ac.il/~gideon`
2. Conceio, A.: A hierarchical genetic algorithm with age structure for multimodal optimal design of hybrid composites. Structural Multidisciplinary Optimization 31, 280–294 (2006)

3. Erlbaum, L.: Conference on Genetic Algorithm, pp. 41–49. Associates Inc., Hillsdale (1987)
4. Glover, F.: Tabu search part I. ORSA Journal on Computing 1(3), 190–206 (1989)
5. Goldberg, D.E.: Genetic algorithms in search, optimization and machine learning. Addison Wesley, Massachusetts (1989)
6. Goldberg, D.E., Richardson, J.: Genetic algorithms with sharing for multimodal function optimization. In: Proceedings of the Second International
7. Hadar, J., Rusell, W.R.: Rules for Ordering Uncertain Prospects. American Economic Review 59, 25–34 (1969)
8. Holland, J.H.: Adaptation in neural and artificial systems. The University of Michigan Press, Ann Arbor (1975)
9. Harik, G.R.: Finding multimodal solutions using restricted tournament selection. In: Eshelman, L. (ed.) Sixth International Conference on Genetic Algorithms, pp. 24–31. Morgan Kaufmann, San Francisco (1995)
10. Jahn, J.: Vector Optimization, Theory, Applications and Extensions, 2nd edn. Springer, Heidelberg (2011)
11. Jensen, E.D.: Topological structural design using genetic algorithms, Phd. Dissertation, Purdue University, Lafayette (1992)
12. Li, J.-P., Balazs, M.E., Parks, G.T., Clarkson, P.J.: A species conserving genetic algorithm for multimodal function optimization. Evolutionary Computation 10(3), 207–234 (2002)
13. Li, J.-P., Li, X.-D., Wood, A.: Species Based Evolutionary Algorithms for Multimodal Optimization. In: A Brief Review, WCCI 2010 IEEE (2010)
14. Kennedy, J., Eberhart, R.C.: Particle Swarm Optimization. In: IEEE Int.Conf. on Neural Networks, pp. 1942–1948 (1995)
15. Kicinger, R., Arciszewski, T., De Jong, K.A.: Evolutionary computation and structural design: a survey of the state of the art. Computers and Structures 83(23-24), 1943–1978 (2005)
16. Kim, I.Y., de Weck, O.L.: Variable chromosome length genetic algorithm for progressive refinement in topology optimization. Structural and Multidisciplinary Optimization 29(6), 445–456 (2005)
17. Kirk Martini, P.E.: Harmony Search Method for Multimodal Size, Shape, and Topology Optimization of Structural Frameworks. Journal of Structural Engineering 137(11), 1332–1339 (2011)
18. Mahfoud, S.W.: Crowding and preselection revisited. In: Bnner, R.M., Manderick, B. (eds.) Proceedings of the Second International Conference on Parallel Problem Solving from Nature - PPSN 2011, vol. 36, pp. 27–36. Elsevier Science Publishers (1992)
19. Mengshoel, O.J., Goldberg, D.E.: The Crowding Approach to Niching in Genetic Algorithm. Evolutionary Computation 16(3), 315–354 (2008)
20. Rechenberg, I.: Evolutionsstrategie: Optimierung technischer systeme nach prinzipien der biologischen evolution, Stuttgart. Fommann-Holzbook (1973)
21. Sandgren, E., Jensen, E.D., Welton, J.: Topological design of structural components using genetic optimization methods. In: Proceedings of the Winter Annual Meeting of the American Society of Mechanical Engineers, pp. 31–43 (1990)
22. Stoean, C., Preuss, M., Stoean, R., Dumitrescu, D.: Disburdening the Species Conservation Evolutionary Algorithm of Arguing with Radii. In: GECCO 2007, pp. 1420–1427 (2007)
23. Wang, S.Y., Tai, K.: Graph representation for structural topology optimization using genetic algorithms. Computers and Structures 82(2021), 1609–1622 (2004)

24. Wang, S.Y., Tai, K.: Structural topology design of optimization using genetic algorithm with a bit-array representation. Computer Methods in Applied Mechanics and Engineering 194(3638), 3749–3770 (2005)
25. Woon, S.Y., Tong, L., Osvaldo, O.M., Steven, G.P.: Effective optimization of continuum topologies through a multi-GA system. Computer Methods in Applied Mechanics and Engineering 194(3033), 3416–3437 (2005)
26. Li, X.: Adaptively Choosing Neighbourhood Bests Using Species in a Particle Swarm Optimizer for Multimodal Function Optimization. In: Deb, K., Tari, Z. (eds.) GECCO 2004. LNCS, vol. 3102, pp. 105–116. Springer, Heidelberg (2004); Fommann-Holzbook 1973