Structural and Multidisciplinary Optimization
https://doi.org/10.1007/s00158-018-1943-1

RESEARCH PAPER

Statistical analysis of control parameters in evolutionary map L-systems-based topology optimization

Teemu J. Ikonen[1] · András Sóbester[1]

Received: 9 November 2017 / Revised: 20 January 2018 / Accepted: 8 February 2018

Abstract

Map L-systems-based parametrization, also referred to as the cellular division method, is a generative encoding, suitable for topology optimization. The parametrization is compact due to its ability to reuse its elements, and therefore capable of covering a large design space with relatively few design variables. Map L-systems are often evolved using genetic algorithms (GAs). A key implementation detail of such procedures, as with most GA-based geometry searches, is the choice of parameters controlling the operation of the evolutionary process. The optimal choice of these in conventional optimization formulations is highly problem-specific—far less so, however, when the GA evolves an L-systems encoding and does not act directly on the geometry. This is because the L-system encoding is, itself, independent of the geometry. We study the effects of different control parameters by conducting a statistical test of over 400 parameter combinations on five test cases, for which the global optima are known. The best-performing parameter combinations are reported as a Pareto front of the average number of objective function evaluations and ranking based on the average of optimized fitnesses. Finally, three Pareto-optimal parameter combinations are selected and applied to an optimization problem of maximizing the fundamental natural frequency of an integrally stiffened aluminum panel. The best of the resulting designs has a higher fundamental natural frequency than the baseline design by a margin of 11.2%.

Keywords Topology optimization · Generative encoding · L-systems · Genetic algorithm · Control parameters

1 Introduction

Evolutionary algorithms are population-based search heuristics that mimic two revolutionary discoveries in biology: Darwinian natural selection and the identification of the deoxyribonucleic acid (DNA) sequence inside the nucleus. The DNA sequence stores the genetic information, i.e. genotype, of a living organism. Instead of explicitly encoding existence of individual cells in the organism, the DNA sequence is a developmental recipe that implicitly constructs the phenotype of the organism.[1] However, the parameterization in evolutionary algorithms is often conducted using design variables that explicitly define units of the phenotype, referred to as *direct encoding*. Implicit parameterizations, referred to as *generative encodings* (also *developmental encodings* and *artificial embryogeny*) define a developmental recipe that produce the phenotype. They have better scalability and are more compact than direct encodings due to their capability of reusing elements of the genotype, which enables the formation of self-similar and hierarchical sub-parts in the phenotype (Hornby and Pollack 2001; Stanley and Miikkulainen 2003; Deaton and Grandhi 2014).

Topology optimization comprises search methods to seek the optimal material distribution in a given design domain. For an extensive review of the topic, the reader

Responsible Editor: Nestor V. Queipo

The electronic supplementary material is also openly available from the University of Southampton repository at https://doi.org/10.5258/SOTON/D0431.

Electronic supplementary material The online version of this article (https://doi.org/10.1007/s00158-018-1943-1) contains supplementary material, which is available to authorized users.

✉ Teemu J. Ikonen
t.j.ikonen@soton.ac.uk

András Sóbester
a.sobester@soton.ac.uk

[1] Computational Engineering and Design, University of Southampton, Southampton, Hampshire, SO16 7QF, UK

[1] Biological phenotypes are, in fact, also dependent on epigenetic and environmental factors.

Published online: 19 March 2018

may wish to consult References Deaton and Grandhi (2014), Bendsøe and Sigmund (2003) and Huang and Xie (2010). Commonly used gradient-based methods, such as SIMP (Solid Isotropic Material with Penalization) and ESO (Evolutionary Structural Optimization), use the direct encoding, where each of the design variables determines the existence/density of a single material element in the phenotype. Thus, even in a two-dimensional design domain, the number of required design variables increases quadratically as a function of the mesh resolution. Another type of direct encoding is the so-called *ground structure* approach, where a dense set of candidate structural members is fitted inside a design domain, and the optimal subset of these members is sought. One of the drawbacks of this approach is that the ground structure must also be defined a priori for each optimization problem. Evolutionary algorithms with a generative encoding provide an alternative, designed to mitigate these issues.

One such encoding applies Lindenmayer systems (Lindenmayer 1968a, b) (L-systems) to develop the phenotype in stages. Stanley and Miikkulainen (2003) categorize the method as a grammatical approach of artificial embryogeny, as they are defined in a language of *formal grammars* (Chomsky 1956). Two common graphical interpretation methods of L-systems are the *turtle interpretation* and *map L-systems*, which were initially developed to model the growth of plants and cellular layers, respectively.

L-systems-based parameterizations have been applied to several topology optimization studies. Hornby and Pollack (2001) applied L-systems, with the turtle interpretation, as a parameterization method to the design search of a table structure. Subsequently, the authors evolved robots for locomotion (Hornby and Pollack 2002), by parameterizing both their body and neural controller using the same methods. In both applications, the authors observed that algorithms with generative encoding yielded designs with higher fitness and were faster than corresponding algorithms with direct encoding. Rieffel et al. (2009) used map L-systems in design optimization of irregular tensegrity structures. Kobayashi (2010) evolved venation patterns of artificial cordate leaves in multi-objective optimization, minimizing both the mass of the leaf and its pressure drop. He also showed that the designs he obtained were robust and fault resistant, in a similar way to their biological counterparts. Pedro and Kobayashi (2011) benchmarked the map L-systems-based encoding against a direct encoding (also driven via an evolutionary algorithm), on a cantilever beam problem. Their results showed that the algorithm with generative encoding yielded designs with similar optimized fitness values using fewer objective function evaluations than the algorithm with the direct encoding. Sabbatini et al. (2015) applied L-systems, with turtle interpretation, to multi-objective stiffener layout optimization, minimizing the vibration amplitude and mass of a plate structure. Allison et al. (2013) included a nested sizing routine to a map L-systems-based algorithm, and applied the algorithm to topology optimization of a truss structure. Stanford et al. (2012, 2013) evolved, using map L-systems, the venation and the mechanism of a flapping wing, to improve its aerodynamic performance. In addition, map L-systems have been applied to various other topology optimization studies on aircraft wings (Kobayashi et al. 2010; Kolonay and Kobayashi 2010, 2015; Ikonen and Sóbester 2017).

The map L-systems-based parameterization has gained popularity among topology optimization researchers, perhaps because map L-systems can conveniently be mapped inside a finite two-dimensional design domain. In the majority of the resulting publications, map L-systems are evolved via a genetic algorithm (GA). Further, several studies (Pedro and Kobayashi 2011; Stanford et al. 2012, 2013; Ikonen and Sóbester 2017; Allison et al. 2013) use similar numerical representations to encode map L-systems into a vector format, which originate from that defined by Pedro and Kobayashi (2011). In spite of the extensive use of evolutionary algorithms to search the space of L-systems encodings, no systematic efforts have been reported to date to understanding the impact of evolutionary algorithm parameter choices on the performance of such optimization processes. The identification of optimal control parameters is a notoriously difficult aspect of evolutionary search heuristic design due to the problem-specific nature of any findings. However, the starting point of this paper is that techniques where the evolutionary process operates on the encoding—such as L-systems based methods—and not directly on the design, are less affected by this problem dependence. The encoding can be seen as an intermediary layer of the problem, which ‘shields’ the evolutionary search from some of the variability resulting from the objective function of the structural design problem.

In this study we perform a statistical experiment involving 432 control parameter combinations on the map L-systems-based topology optimization method, using the numerical representation proposed by Pedro and Kobayashi (2011). We conduct the experiments on five simple optimization problems, with known global optima. The goal is to design a search that yields a good objective function value in a small number of objective function evaluations. As these performance measures are often competing, we report our results as a Pareto front of the two. In addition, we examine whether, or to what extent, the rankings of parameter combinations, based on the optimized objective function value and the required number of function evaluations, are problem-dependent.

Finally, we use a Pareto-optimal parameter combination in the topology optimization of an integrally stiffened aluminum panel, in order to demonstrate the potential impact of the technology in a ‘real-life’ engineering context. The objective is to maximize the lowest natural frequency of the structure, subject to a mass constraint. The obtained design is compared against corresponding results with iso- and orthogrids, often seen in aerospace and automotive applications.

2 Map L-systems

L-systems are language-theoretic models for the development process of living organisms (Rozenberg and Salomaa 2012). The original purpose of L-systems was to mathematically model the developmental process of a living organism, with a particular emphasis on its topology. An example area of application has been the modeling of plants, in which case the topology defines how its substructures, such as the trunk, branches, and leafs, are aligned with respect to each other. More recently, L-systems have been applied to a variety of other fields too, such as computer graphics, artificial intelligence and engineering.

In this study, we use map L-systems, a variation of L-systems, which enables the production of mapped topologies, such as cellular layers. Here, a map is defined to consist of a finite set of regions, i.e. cells, which are described by a circular sequence of edges, having a finite length (Prusinkiewicz and Lindenmayer 2012). More precisely, we use *Binary Propagating Map OL-systems with markers* (mBPMOL-systems), as proposed by Nakamura et al. (1986). The system is binary because, during a cell division, each cell can only split into two offspring cells. The word ‘propagating’ defines that, once created, the edges cannot be removed, and therefore the cells cannot fuse or die. The letter ’O’ indicates that the cell divisions are context-free, which means that cells do not interact with each other.

Like other types of L-systems, an mBPMOL-system consists of an alphabet Σ, axiom ω_0, and N_P rewriting rules $P_1, \ldots P_{N_\mathrm{P}}$. The main idea is that the axiom, defined as a sequence of letters from the alphabet, is modified iteratively n times based on the rewriting rules, where n represents the age of the organism. The resulting topology after n iterations is referred to as the nth *developmental stage* of the system.

To illustrate the process, let us consider a map L-system (described, for example, by Prusinkiewicz and Lindenmayer (2012)), for which the alphabet is defined as $\Sigma \equiv \{A, B\}$, and the axiom as $\omega_0 = ABAB$. The axiom is mapped into a single cell, having edges corresponding to the letter sequence (Fig. 1a). The edges are ordered clockwise starting from the bottom edge. Further, the following two rewriting rules are defined for the letters A and B in the alphabet:

$$\begin{aligned} &P_1 : A \rightarrow B[-A][+A]B \\ &P_2 : B \rightarrow A \end{aligned} \tag{1}$$

The left- and right-hand sides of the rewriting rules are referred to as the *predecessor* and the *successor*, respectively. A characteristic feature for mBPMOL-systems is the existence of markers in the successor, which are indicated by square brackets. They act as start and end points for new edges, which split cells.[2] The content between the square brackets contains the side of the marker, ‘+’ or ‘−’, and a letter that is referred to as the *label*.

The transition to the next developmental stage includes two phases. In the first, the rewriting rules are applied to all edges in the current developmental stage. The predecessor edge is divided into equally-sized successor edges (Fig. 1b). The markers are assigned to the nodes with respect to their relative location in the successor (cf. the arrows in Fig. 1b). Sides ‘+’ or ‘−’ indicate the location of the marker, which is either the left or right side of the edge, respectively. In the second, matching marker pairs are connected inside each cell, creating new edges to the system and dividing cells into two offspring cells (Fig. 1c). A marker pair is considered to be matching if they are located inside the same cell and have the same label.[3] Only the first matching marker pair inside each cell is connected, and the remaining markers are discarded. The subsequent developmental stages are generated by repeating the same procedure (Fig. 1d–f).

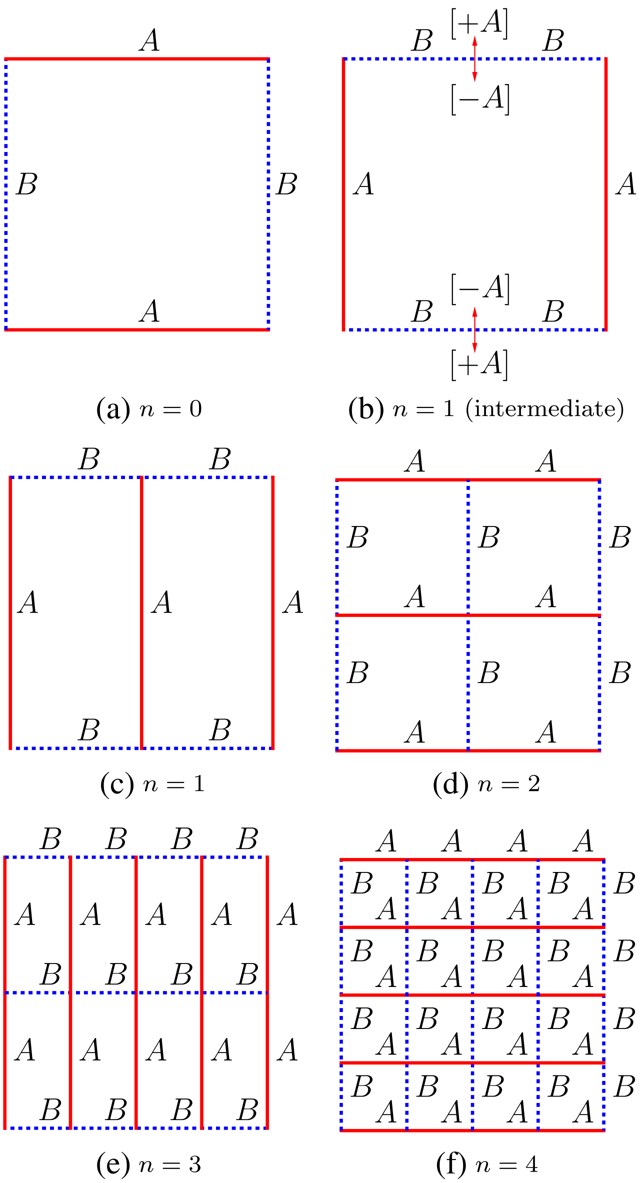
Later in this study we use directional markers. Possible directions for the markers are ‘←’, ‘→’ or neutral, denoted over the marker label, e.g. $[-\overrightarrow{B}]$. The criteria defined above for matching markers is amended by the following: the direction of the start marker must be ‘→’ or neutral, and the direction of the end marker ‘←’ or neutral. For simplicity, all edges are defined to have a uniform direction.

The map L-systems may be amended by a dynamic method (Prusinkiewicz and Lindenmayer 2012), where an osmotic pressure is applied inside the cells and an equilibrium state is determined for the vertex locations of the edges, which have a finite axial stiffness coefficient (Pedro and Kobayashi 2011) included the method in their design space parameterization). However, the method requires solving the equilibrium stage iteratively at every developmental stage. We omit the dynamic method from the parameterization, as we need to keep the computational cost low to allow us to perform a large number of experiments.

The developmental stages in Fig. 1 are only a few example topologies that may be generated by mBPMOL-systems. A diverse range of different topologies may be generated by varying the axiom and the rewriting rules, and

[2] Markers have a counterpart in biology, preprophase bands of microtubes (Prusinkiewicz and Lindenmayer 2012).

[3] When map L-systems are used as a parameterization method in topology optimization, additional requirements may be considered, such as a minimum fraction of the offspring cell area in comparison to the parent cell area (cf. for example reference Pedro and Kobayashi 2011).



(a) $n = 0$

(b) $n = 1$ (intermediate)

(c) $n = 1$

(d) $n = 2$

(e) $n = 3$

(f) $n = 4$

Fig. 1 The axiom ($n = 0$) and first four developmental stages ($n = 1 \ldots 4$) of the example map L-system. Subfigure (**b**) shows the intermediate stage before joining the matching markers. Edges A and B are indicated by continuous blue and dashed red lines, respectively

further by including more letters in the alphabet. In the next section, we describe how the topologies are evolved using a genetic algorithm. In the remainder of this paper, we will refer to the mBPMOL-systems simply as map L-systems.

2.1 Evolution via a genetic algorithm

Evolutionary algorithms are population-base optimization heuristics that mimic Darwinian natural evolution. In this simulated evolution paradigm a merit function (objective function) takes the role of the environment, and mathematical transformations are defined to represent reproduction, crossover and mutation.

As described in the introduction, the map L-systems-based parameterization is commonly evolved via GAs, and, recently, with numerical representations similar to the one by Pedro and Kobayashi (2011). The purpose of their representation is to encode a map L-system, consisting of the axiom, rewriting rules and additional variables, as a vector of real numbers in the range [0, 1]. It is designed to prevent the formation of invalid features, such as markers with multiple labels or an axiom with non-alphabetic characters. The design space of the representation is determined by the length of the axiom, N_{a}, the number of encoded rewriting rules, N_{P}, and N_{v} additional variables. The number of encoded rewriting rules is equal to the number of letters in the alphabet Σ. Details of the numerical representation used in this work, containing minor modifications to the one by Pedro and Kobayashi (2011), are described in Appendix A. The genetic algorithm is implemented in Python, using Pyevolve (Perone 2009), an open-source library of genetic operators.

3 Statistical analysis of control parameters

There is, in general, a strong relationship between the choice of the control parameters of a GA and its effectiveness (its ability to find good solutions) and efficiency (its ability to find them quickly).

In this section we conduct a series of statistical experiments to quantify this relationship in the case of the map L-systems-based topology optimization method. In what follows, we describe the experimental plan (Section 3.1) and the test cases (Section 3.2). We then deploy these to gain an empirical understanding of the performance of the algorithm with a range of parameter choices (Section 3.3).

3.1 Experimental plan

Table 1 reviews the control parameters used in the literature (where specified) in L-systems-based topology optimization. We list the following parameters:

- selection strategy
- tournament pool size N_{pool}
- crossover and mutation types
- crossover rate c_{x}
- mutation rates c_{M} and c_{m}
- elitism E_{bool}
- termination criteria.

Mutation rate c_{m} is the element-specific probability of mutation, applied to a c_{M} proportion of the population. In

Table 1 Survey of control parameters in L-systems-based topology optimization studies found in the literature

Publication	Selection	Population size N_{pop}	Crossover	Mutation	Elitism E_{pool}	Termination
Rieffel et al. (2009)	Roulette wheel	100	One-point crossover between rules ($c_x = 0.2$)	Primary and secondary ($c_M = 0.4$ each)[a]	True	Terminated after 500 generations
Kobayashi (2010)[b]	Tournament ($N_{\text{pool}} = 4$)	200	Distributed ($c_x = 0.8$)	Gaussian distributed ($c_M = 0.2$)[c]	True	Terminated after 100 generations
Pedro and Kobayashi (2011)	Stochastic universal sampling (Baker 1987)	50–100	Distributed ($c_x = 0.8$)	Gaussian distributed ($c_M = 0.15$)	True	Terminated after 100 generation, or after 50 generations without improvements
Sabbatini et al (2015)[b]	Tournament ($N_{\text{pool}} = 4$)	100	Distributed ($c_x = 0.8$)	Gaussian distributed[c] ($c_M = 0.19$)	True	Terminated after 100 generations
Ikonen and Sóbester (2017)	Tournament ($N_{\text{pool}} = 4$)	150	Two-point ($c_x = 0.8$)	Swap ($c_M = 1.0$, $c_m = 0.02$)	True	Terminated after 30 generations without improvements

[a] Primary mutation randomly selects one of the rules and randomly changes its right side. Secondary mutation makes a small change to the resulting map L-system

[b] In addition, the author(s) applies inversion to the axiom letters, with a probability of 0.01

[c] Gaussian distributed random mutation is added to all elements. The random distribution has a zero mean and a decreasing standard deviation as a function of the prevailing generation. The mutation is applied to the individuals, to which crossover was not applied

References Kobayashi (2010) and Sabbatini et al. (2015) the generation of phenotype follows the turtle interpretation (Prusinkiewicz and Lindenmayer 2012). However, the turtle interpretation of L-systems is still a generative encoding, specifying the phenotype via the axiom, rewriting rules and additional variables, and therefore we have included it in the review. As Table 1 shows, the variation in parameters across the selection of studies we were able to gather is significant. The only exception is whether elitism was used, E_{bool}, which was 'True' in all studies. We were not able to find any studies that provided a clear reasoning behind their particular choice of parameters.

To study the effects of choosing a particular set of control parameters, and to find suitable parameter combinations, we ran a statistical experiment on five simple test cases, presented in Section 3.2. Table 2 shows our design of experiments. Of the two mutation rate types prevalent in the studies reported so far on L-systems based optimization, c_M and c_m, our experiments vary the latter, keeping the former fixed at $c_M = 1.0$. We tested all 432 control parameter combinations 70 times on each of the five test cases. The optimization runs were terminated when no improvements were obtained during 30 consecutive generations. We used a mutation operator that swaps two randomly selected elements in an individual, and the number of elite individuals, when applicable, was set to one. Optimization runs were initiated from a population of random individuals.

Performance of GAs may be improved, in many cases, by seeding the initial population with a diverse set of decent initial guesses (cf. for example Reference Simpson and D'souza 2004). However, in the vast majority of studies, using an L-systems-based parameterization, optimization

Table 2 Control parameter values of the statistical experiment

Parameter	Values
Population size N_{pop}	{50, 100, 150, 200}
Pool size N_{pool}	{2, 4, 8}
Crossover rate c_x	{0.6, 0.8, 1.0}
Mutation rate c_m	{0.0, 0.02, 0.04}
Crossover type X_{type}	{two-point, distributed}
Elitism E_{pool}	{True, False}

All parameter combinations, totalling 432, are tested separately

Table 3 Definition of the L-system design space

Parameter	Values
Axiom length N_a	4
Number of rewriting rules N_P	4
Number of tokens N_r	6
Minimum area fraction f_a	$0\dots0.5$
Age n	$1\dots6$

Minimum area fraction f_a and age n are additional variables

runs are initiated from a random population.[4] Finding a technique to define these initial guesses with sufficient diversity for the L-systems-based parameterization falls outside of the scope of the current study.

3.2 Test cases

This section defines the five test cases. Their objective functions are based on the geometric features of a map L-system. The test cases have low computational cost and known global optima. Before going into their details, let us define the design space at the L-system level, which is kept constant throughout the experiment.

The parameters defining the design space are listed in Table 3. The axiom of map L-systems is mapped as a unit square, and thus the axiom length $N_\mathrm{a} = 4$. Two additional variables are used: f_a defines the minimum fraction between offspring and parent cell areas, and n is the age of the system, i.e. the ordinal of the desired developmental stage.

3.2.1 Test Case 1

The first test case is inspired by the example map L-system described in Section 2. Its goal is to find a map L-system that produces the map of the fourth developmental stage of the example system (Fig. 1f). Thus, the sought map has 16 equally-sized, square-shaped cells. We use here the same objective function as in Reference Ikonen and Sóbester (2017), defined as

$$f_1 = \frac{1}{10}|16 - N_\mathrm{cells}| + S_\mathrm{e} + S_\mathrm{a}, \tag{2}$$

where N_cells is the number of cells, and S_e and S_a are the standard deviations of edge lengths and cell areas in the system, respectively. The optimization processes of two representative runs are presented in Fig. 2. The first run yields the global optimum of $f_1 = 0$, whereas the optimized result of the second run is sub-optimal.

3.2.2 Test Case 2

The second test case is a variation of the first test case. Its goal is to find a map consisting of square-shaped cells of any size. The objective function is defined as

$$f_2 = \frac{1}{2}P_\mathrm{N} + 2\frac{\sum_{i=1}^{N_\mathrm{cells}} s_{\mathrm{e},i}}{N_\mathrm{cells}} + \frac{1}{100}\frac{\sum_{i=1}^{N_\mathrm{cells}} s_{\alpha,i}}{N_\mathrm{cells}}, \tag{3}$$

where $s_{\mathrm{e},i}$ and $s_{\alpha,i}$ are the standard deviations of the edge lengths and cell areas, respectively, of the ith cell in the map. P_N, defined as

$$P_\mathrm{N} = \begin{cases} 5 - N_\mathrm{cells}, \text{ if } N_\mathrm{cells} < 5 \\ 0, \text{ if } N_\mathrm{cells} \geq 5 \wedge N_\mathrm{cells} \neq 16 \\ 1, \text{ if } N_\mathrm{cells} = 16, \end{cases} \tag{4}$$

is a penalty coefficient designed to prevent the optimization from converging to trivial solutions of maps containing 1 or 4 equally-sized cells, or to the global optimum of Test Case 1. While Test Case 1 has a single global optimum, Test Case 2 admits multiple global optima ($f_2 = 0$), (as do Test Cases 3-5). An example global optimum, produced by an optimization process, is shown in Fig. 3a.

3.2.3 Test Case 3

The purpose of the third test case is to minimize the fraction of the number of nodes, N_nodes, with respect to the number of cells, N_cells, in the map. Thus, the objective function is defined as

$$f_3 = \frac{N_\mathrm{nodes}}{N_\mathrm{cells}}. \tag{5}$$

Let us derive the values of the global optima. First of all, the global optima are maps consisting exclusively of triangles. The reason is that the objective function f_3 of a map, containing a polygon with four or more vertices can always be decreased by dividing the polygon into two or more triangles. Based on Euler's formula for planar graphs, and assuming that the map exclusively consists of triangles, the number of cells

$$N_\mathrm{cells} = 2N_\mathrm{nodes} - B_\mathrm{nodes} - 2, \tag{6}$$

where B_nodes is the number of nodes laying at the convex boundary of the graph.[5] The equation can be rewritten as

$$\frac{N_\mathrm{nodes}}{N_\mathrm{cells}} = \frac{1}{2} + \frac{B_\mathrm{nodes} + 2}{2N_\mathrm{cells}}. \tag{7}$$

Therefore, the objective function f_3 (5) reaches the global minimum, when B_nodes and N_cells reach their minimum and maximum, respectively. The minimum number of boundary nodes, B_nodes, is equal to number of nodes in the map

[4] An exception is the study by Kobayashi (2010), where an optimization process is initiated from the final population of another optimization process with a slightly different objective function.

[5] All developmental stages of map L-systems, initiated from an axiom mapped onto a unit square, have a convex boundary, if the dynamic method (Prusinkiewicz and Lindenmayer 2012) is not used.

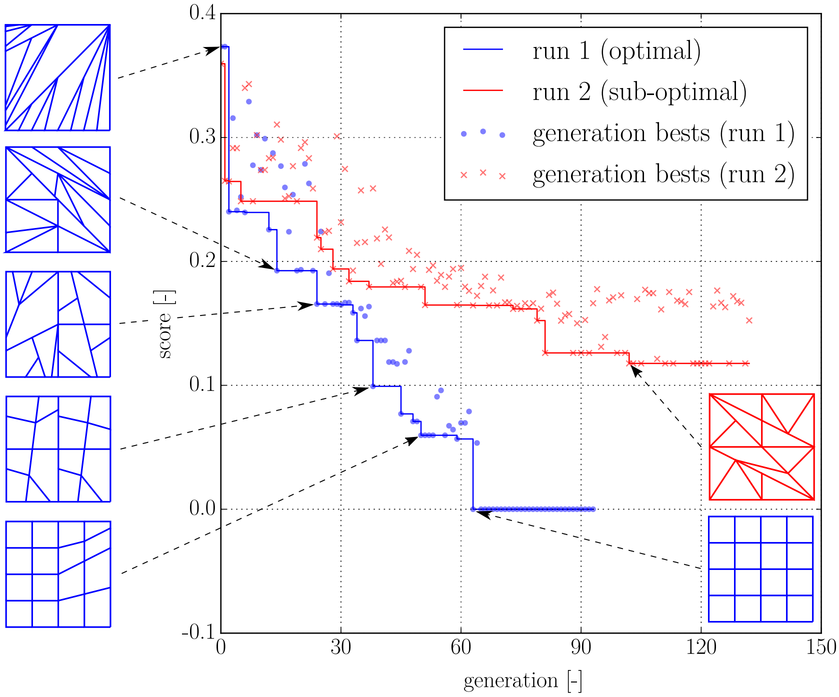


Fig. 2 Representative optimization runs yielding the global optimum and a sub-optimal final result

corresponding to the axiom, i.e. $B_{\text{nodes}} = 4$. On the other hand, the maximum age n is defined to be 6. As the number of cells at most doubles at every developmental stage, the maximum number of cells N_{cells} is $2^6 = 64$. Thus, the global optimum is $f_3 = \frac{35}{64}$. An example global optimum is shown in Fig. 3b. It is noticeable that the boundary of the map only includes the four nodes related to the axiom.

3.2.4 Test Case 4

An *m-equidissection* of a polygon is set of m non-intersecting triangles, having an equal area and whose union is the polygon. The purpose of the fourth test case is to find a 12-equidissection of the unit square, using an objective function defined as

$$f_4 = \frac{1}{2}|12 - N_{\text{cells}}| + 10S_{\text{a}} + \frac{N_{\text{cells}} - \hat{N}_{\text{cells}}}{N_{\text{cells}}}, \tag{8}$$

where $\hat{N}_{\text{cells}}$ is the number of triangular cells in the map. An example global optimum ($f_4 = 0$) shown in Fig. 3c.

3.2.5 Test Case 5

The fifth test case is a search for a map containing a regular pentagon, filling at least 25% of the unit square. If a pentagon exists in the map, the objective function is defined as

$$f_5 = P_{\text{A}} + \frac{1}{100}s_{\alpha,k} + s_{\text{e},k}, \tag{9}$$

else $f_5 = 10$. P_{A} is a penalty coefficient defined as

$$P_{\text{A}} = \begin{cases} \frac{1}{4} - A_k, \text{ if } A_k < \frac{1}{4} \\ \quad 0, \text{ if } A_k \geq \frac{1}{4}, \end{cases} \tag{10}$$

where A_k is the area of the largest pentagonal shaped cell k. Further, $s_{\alpha,k}$ and $s_{\text{e},k}$ are the standard deviations of the edge angles (in degrees) and edge lengths, respectively, of the cell k. The global optimum has the value of $f_5 = 0$, though this was not found during the experiments. The design with the lowest objective function value is shown in Fig. 3d.

3.3 Results and discussion

The statistical experiment was performed in parallel, using 128 Central Processing Units (CPUs). The total wall time of the experiment was around 15 days.

GAs, characterized by the parameter combinations from Table 2, are applied to Test Cases 1–5, each run 70 times. Global optima were found for Test Cases 1–4 (see Figs. 2 and 3a–c). The lowest obtained objective function value for Test Case 5 ($f_5 = 2.72 \cdot 10^{-2}$) was encountered once among the optimized designs. Although the corresponding design (Fig. 3d) is not the global optimum, it contains a cell that is very close to a regular pentagon and fills more than 25% of the unit square area. The statistical experiment included a considerably large number of optimization runs on each test case: 30240 (70 repeats with 432 parameter combinations).

Let us first examine the results as a series of scatter plots (Fig. 4) of the average number of objective function

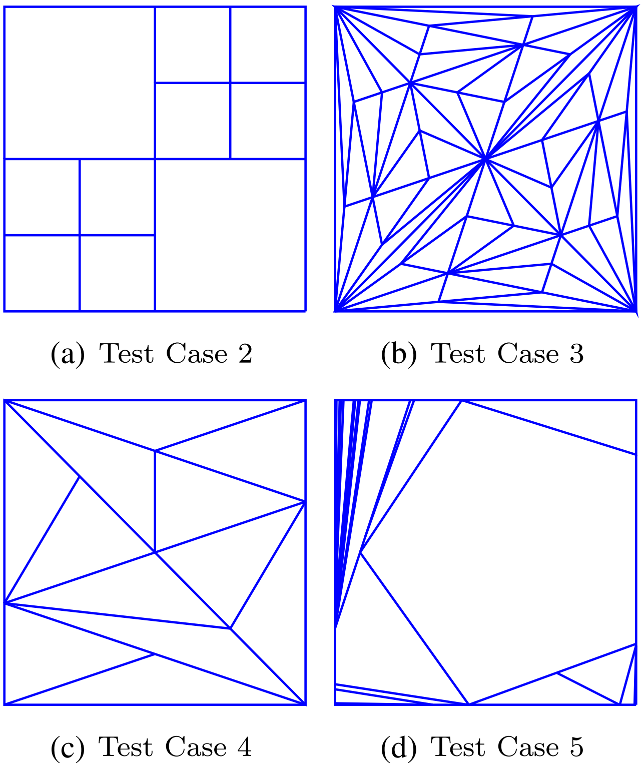


Fig. 3 Example global optima in Test Cases 2–4, and the design with the lowest objective function value in Test Case 5

evaluations, $\bar{Q}$, and the completion rate in terms of finding the global optimum, p_c. As $p_c = 0$ for all control parameter combinations on Test Case 5, we exclude its results from the scatter plots. Thus, a point in the scatter plot represents an average of 280 optimization runs (4 test cases with 70 repetitions). Each subplot in the scatter plot shows the effect of the variation of control parameters on the performance of the GA. The Pareto front of the two objectives is marked by the dashed line.

The population size N_{pop} has the clearest influence on the performance of the GA (Fig. 4a). The points are aligned into bands, approximately parallel to the abscissa, according to their value of N_{pop}. It can clearly be seen that the larger N_{pop} is, the more objective function evaluations are required, but also more likely the GA is to find the global optimum. All population sizes are represented on the Pareto front.

The pool size N_{pool} (Fig. 4b) and the element-wise mutation rate c_m (Fig. 4d) have a similar influence on the performance of the GA. Both of these parameters were tested with a range of three values, with the lowest, $N_{\text{pool}} = 2$ and $c_m = 0.0$, clearly showing the poorest performance. Almost the entire Pareto front is populated by the highest values, $N_{\text{pool}} = 8$ and $c_m = 0.04$. The relative performance differences between these two parameters seem independent of the population size N_{pop}.

The two values ($True/False$) for the elitism E_{bool} (Fig. 4f) divide the four bands of population sizes each into two subbands, again approximately parallel to the abscissa of the plot. The value $E_{\text{bool}} = True$, represented by the upper subband, extends slightly further to the positive direction of the abscissa, and its points form most of the Pareto front.

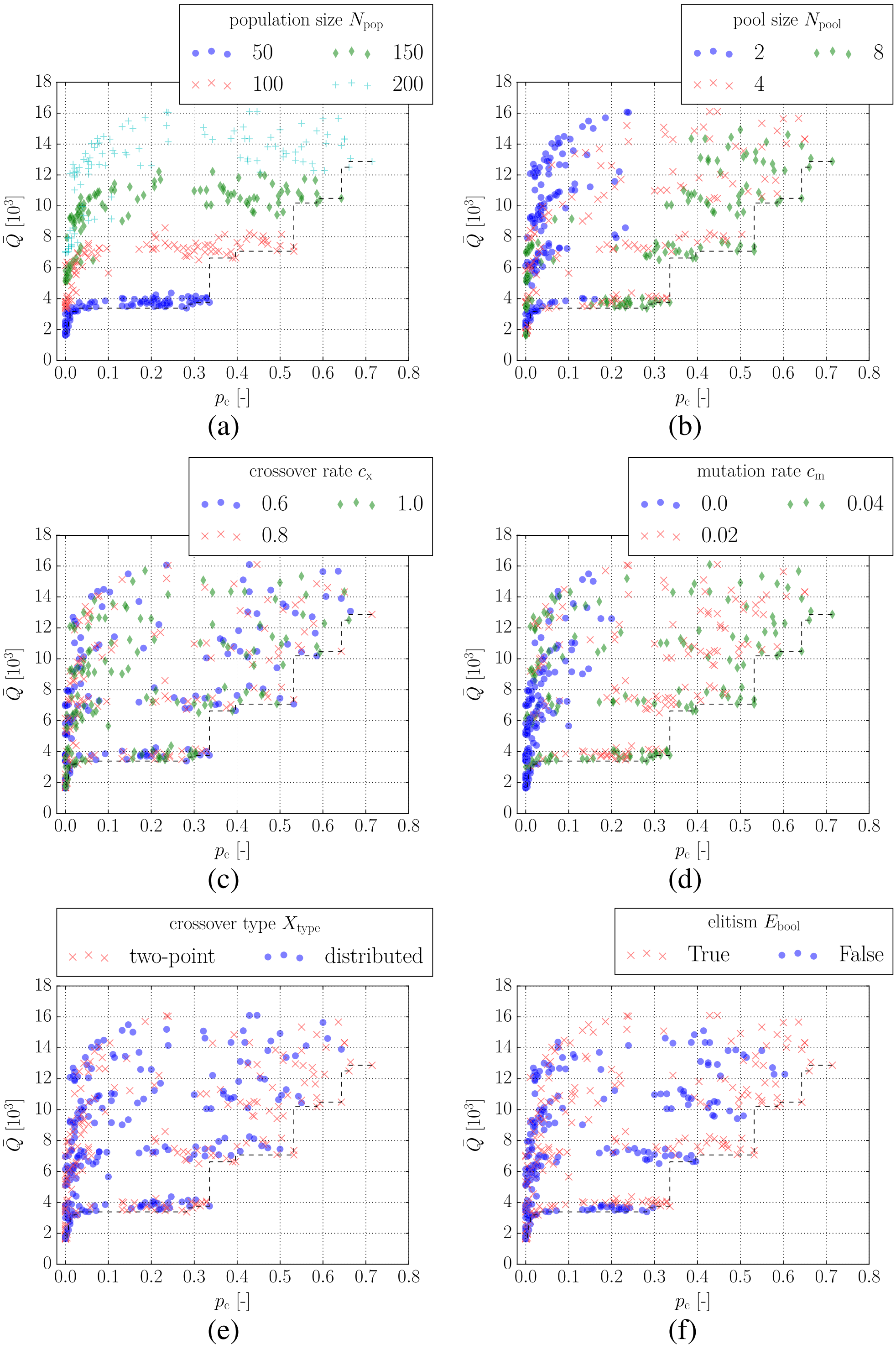
The two-point crossover provides, on average, slightly better completion rate than the distributed crossover (Fig. 4e), and its points form most of the Pareto front. However, the performance difference between the crossovers is small. The crossover rate c_x (Fig. 4c) has very little influence on the performance of the GA (compared to the other tested parameter values), as its parameters are scattered inside the cloud of points.

The completion rate as a performance measure has a drawback. It cannot rank two optimized designs if they both are sub-optimal, and therefore some of the information generated by the experiment is discarded. An alternative may be to directly compare the minimized objective function values. This metric also allows the inclusion of incomplete searches (such as our fifth test case) in the analysis. Since the minimized objective function values are not comparable across test cases, we use rankings as a means of direct comparison. First, the control parameter combinations are ranked, separately in each test case, based on the average minimized objective function value attained by the GA run with each. Second, the obtained ranks are averaged and these values are used as a performance measure.

Figure 5 shows scatter plots using the average rank as a performance measure, along with the average number of objective function evaluations. The broad trends are similar to those seen in the completion rate (Fig. 4), although less pronounced. Let us now extract the Pareto front (dashed line), in the space of minimum average number of objective function evaluations versus minimum average rank, into Table 4. The listing of non-dominated control parameter combinations is ordered from the lowest average rank to the highest. It is noticeable that the population size N_{pop} sweeps through its tested range $N_{\text{pop}} = \{50, 100, 150, 200\}$, in the opposite order, along the Pareto front. These 20 Pareto-optimal combinations selected from the set of 432 tested control parameter combinations, can be viewed as prime candidates when selecting the parameters of a GA to be deployed on a not yet seen problem.

Depending on the budget available for experimentation on the 'real' problem, the analyst may choose to narrow down the list further. First, combination #12 may be considered a practical limit, as points below it provide very marginal decrease in the average number of objective function evaluations as a return of the sacrificed average rank. Second, if we assume that the modality of Test

Fig. 4 Results of the statistical experiment, in terms of the completion rate p_c and the average number of required objective function evaluations $\bar{Q}$. Test Case 5 is excluded from the results, as its global optimum was not found. The dashed line represents the Pareto front between minimum average number of function evaluations and maximum completion rate



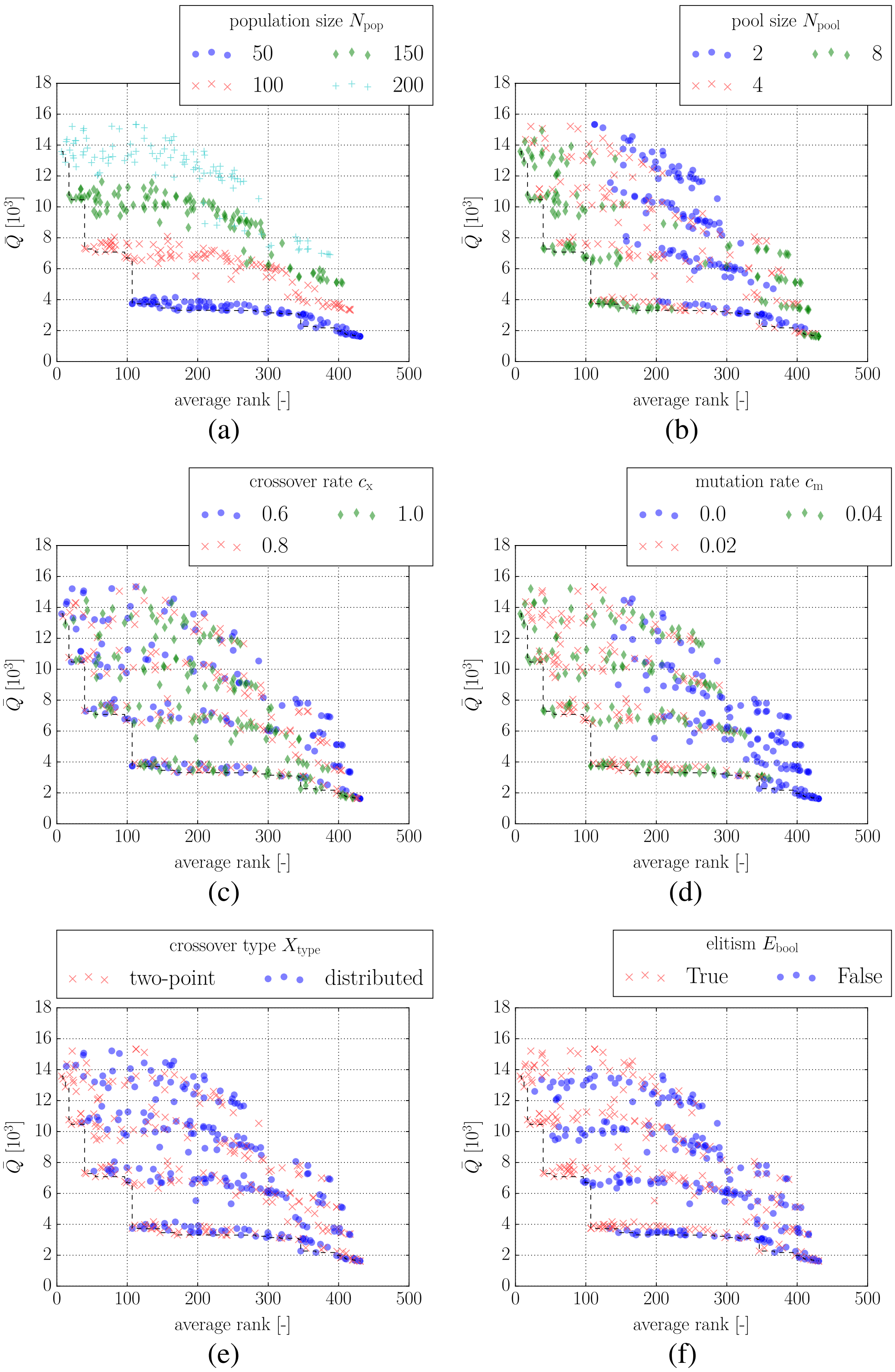
Cases 1–4 is representative of the problem being tackled, there is another way in which the remaining options can be narrowed. The probability p_{global} of finding the global optimum, by performing multiple optimization runs, is defined as

$$p_{global} = 1 - (1 - p_c)^m, \tag{11}$$

where $m \in \mathbb{N}$ is the number of repeated optimization runs. Let us fix $p_{global} = 0.95$, and find the parameter combination at the Pareto front that has the smallest estimate of required objective function evaluations

$$Q_g = \bar{Q}m, \tag{12}$$

Fig. 5 Results of the statistical experiment using the average rank. The parameter combinations are ranked based on their average minimized objective function value. The dashed line represents the Pareto front between minimum average number of function evaluations $\bar{Q}$ and minimum average rank



(a) (b) (c) (d) (e) (f)

where $m = \log_{(1-p_c)}(0.05)$, rounded to the next natural number. The smallest $Q_g (= 29.12 \cdot 10^3)$ is obtained by combination #7 in Table 4, and corresponds to four repeated runs. As a comparison, combination #1, having the smallest average rank, requires only three repeated runs, but these runs require on average more objective function evaluations, and therefore $Q_g = 40.74 \cdot 10^3$.

Parameter combination #7 is, broadly, in keeping with common quidelines for formulated in the general GA literature. However, the tournament pool size N_{pool} =

Table 4 The parameter combinations lying on the Pareto front in Fig. 5

#	N_{pop}	N_{pool}	c_{x}	c_{m}	X_{type}	E_{bool}	Ranks (in test cases)					Average rank	p_{c}[a] [-]	$\bar{Q}$ [10^3]
							1	2	3	4	5			
1	200	8	0.6	0.04	Two-point	True	4	20	5	2	3	6.8	0.6643	13.58
2	200	8	0.8	0.04	Two-point	True	2	9	1	24	8	8.8	0.7143	13.40
3	200	8	1.0	0.04	Two-point	True	1	6	11	23	20	12.2	0.6607	12.89
4	150	8	0.8	0.04	Two-point	True	8	7	6	51	14	17.2	0.6429	10.76
5	150	8	0.6	0.04	Two-point	True	9	34	17	37	6	20.6	0.5857	10.46
6	100	8	0.8	0.04	Two-point	True	57	51	24	29	36	39.4	0.5321	7.34
7	100	8	0.6	0.04	Two-point	True	50	69	15	61	11	41.2	0.5321	7.28
8	100	8	1.0	0.04	Two-point	True	24	40	56	77	77	54.8	0.4929	7.08
9	100	8	0.6	0.02	Two-point	False	120	143	93	95	31	96.4	0.2786	6.77
10	100	8	0.8	0.02	Two-point	False	99	147	117	99	32	98.8	0.2821	6.69
11	50	8	0.6	0.04	Two-point	True	87	111	113	177	47	107.0	0.3143	3.75
12	50	8	0.6	0.04	Distributed	True	105	132	99	136	64	107.2	0.3357	3.73
13	50	8	1.0	0.04	Distributed	True	147	101	90	192	89	123.8	0.2893	3.71
14	50	8	1.0	0.02	Two-point	True	122	176	146	166	65	135.0	0.2357	3.71
15	50	8	1.0	0.02	Two-point	False	137	175	157	127	136	146.4	0.2321	3.46
16	50	8	0.8	0.02	Two-point	False	139	178	147	174	196	166.8	0.2071	3.40
17	50	8	0.6	0.04	Two-point	False	138	145	101	230	242	171.2	0.2821	3.31
18	50	4	0.6	0.04	Two-point	False	173	160	218	314	343	241.6	0.1429	3.30
19	50	4	0.8	0.04	Distributed	False	247	256	278	313	318	282.4	0.0536	3.23
20	50	2	0.6	0.02	Two-point	False	326	302	319	331	316	318.8	0.0214	3.09

The combinations are listed in increasing order of the average rank. An extended version of the table, including all tested parameter combinations, is attached as a supplementary data file

[a] Test Cases 1–4

8 and mutation rate $c_{\mathrm{m}} = 0.04$ may be considered relatively high. Often used values for these parameters are a tournament pool size of 2 (Blickle and Thiele 1995) (or 4) and a mutation rate of 0.005 to 0.01 (Mitchell 1998). In comparison to the general guidelines, the larger tournament pool increases the selective pressure of the evolutionary process, while the increased mutation rate enhances its ability to avoid converging to local optima.

Finally, let us examine the correlation of parameter combination ranks in the five test cases. These ranks are listed in Table 4 for the Pareto-optimal parameter combinations (using the average minimized objective function value as the ranking measure). As a measure, we use Spearman's rank correlation coefficient ρ (Corder and Foreman 2014), which compares the relationship of ordinal or rank-ordered variables. If $\rho = 1$, the correlation is perfect, i.e. the parameter combination ranks are the same among the two test cases. If $\rho = -1$, the correlation is also perfect but the ranks are the opposite. On the other hand, if $\rho = 0$, the ranks are completely independent.

Table 5a and b show the matrices of pairwise correlations of ranks between the five test cases, using the average minimized objective function value and the average number of objective function evaluations, respectively, as ranking measures. The diagonal elements of the matrix are trivial as the comparison is made on the same ranks, obtained from the same test case ($\rho = 1$). Excluding the diagonal elements, the correlation coefficients vary from 0.645 to 0.979, indicating strong correlations between the obtained ranks. This indicates that a parameter combination performing well on one test case is also likely to perform well on another test case.

There is little consistency in the literature in terms of the number of encoded rewriting rules, N_{P}. Our goal here is not to determine the optimal value for N_{P}; rather, we are interested in how sensitive our results, described above, are to variations in N_{P}. To study this, we run experiments with a range of $N_{\mathrm{P}} = \{2 \ldots 6\}$ on Test Case 1. As the optimization runs were repeated 70 times with $N_{\mathrm{P}} = 4$ earlier, we performed the same number of repeats with the other values. The obtained pairwise correlations of ranks between different values of N_{P} are listed in Table 6a and b, using the same ranking measures as in Table 5a and b, respectively. The correlation coefficients, varying from

Table 5 Pairwise Spearman's rank correlation coefficients ρ between the test cases

	ρ	Test Case				
		1	2	3	4	5
(a)						
Test Case	1	1.000	0.968	0.965	0.850	0.727
	2	0.968	1.000	0.949	0.846	0.645
	3	0.965	0.949	1.000	0.897	0.786
	4	0.850	0.846	0.897	1.000	0.841
	5	0.727	0.645	0.786	0.841	1.000
(b)						
Test Case	1	1.000	0.970	0.955	0.953	0.949
	2	0.970	1.000	0.975	0.973	0.934
	3	0.955	0.975	1.000	0.979	0.950
	4	0.953	0.973	0.979	1.000	0.960
	5	0.949	0.934	0.950	0.960	1.000

The ranks are ordered based on the average minimized objective function value (a) and average number of objective function evaluations (b)

0.800 to 0.988, show strong correlation in the ranks obtained with different numbers of rewriting rules, N_P. This indicates that no radical changes are to be expected in the relative performance of parameter combinations if the number of rewriting rules is changed.

The goal of this paper is to offer practitioners of GA-driven L-Systems-based topology search advice on optimization setup, firmly grounded in empirical observations based on a set of test problems. In the next section we tackle a structural geometry optimization problem using an L-systems based heuristic, demonstrating how the results of the empirical study presented above can be implemented in a ‘real-life’ engineering context.

Table 6 Pairwise Spearman's rank correlation coefficients ρ between a range rewriting rules $N_\mathrm{P} = \{2 \dots 6\}$ on Test Case 1

	ρ	N_P				
		2	3	4	5	6
(a)						
N_P	2	1.000	0.956	0.894	0.848	0.800
	3	0.956	1.000	0.956	0.921	0.880
	4	0.894	0.956	1.000	0.976	0.955
	5	0.848	0.921	0.976	1.000	0.977
	6	0.800	0.880	0.955	0.977	1.000
(b)						
N_P	2	1.000	0.975	0.914	0.864	0.847
	3	0.975	1.000	0.960	0.923	0.902
	4	0.914	0.960	1.000	0.981	0.967
	5	0.864	0.923	0.981	1.000	0.988
	6	0.847	0.902	0.967	0.988	1.000

The ranks are ordered in Subfigures a and b using the same measures as in Table 5a and b, respectively

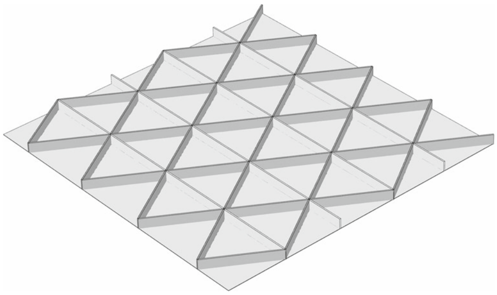


Fig. 6 Integrally stiffened isogrid panel

4 Application: integrally stiffened aluminum panel

The results we obtained from the simple test cases showed that L-systems can generate diverse topologies containing triangular, quadrilateral and pentagonal geometries. In structural engineering, iso- and orthogrids are integrally stiffened panel structures, with topologies also consisting of triangular and quadrilateral geometries,[6] respectively (Huybrechts et al. 2002). Figure 6 illustrates the stiffener topology of an isogrid structure. These structures are seen in many aircraft and satellite structures.

Considering metal structures, integrally stiffened panels are constructed using subtractive manufacturing techniques, where material is successively removed from a solid piece of material, using methods like face milling or chemical etching. These panels are free of attachment flanges and rivet holes, which enables lighter designs, better damage tolerance and cheaper manufacturing in comparison to panels constructed via traditional manufacturing techniques, such as riveting (El-Soudani 2006).

The purpose of this section is to apply the map L-systems based topology optimization method to the design search of the optimal stiffener topology in an integrally stiffened aluminum panel, and to benchmark the results against corresponding iso- and orthogrid structures, for which the optimal stiffener spacing is sought. The map L-systems-based parameterization is well-suited for the purpose because it produces two-dimensional phenotypes that can be directly used to describe stiffener topologies of design candidates.

[6] More precisely, these geometries are equilateral triangles and squares.

Table 7 Properties of the optimization problem

Property	Value
Panel dimensions	1 m × 1 m
Total mass	8 kg
Panel thickness	1 mm
Stiffener aspect ratio	7.5 [−]
Elastic modulus	73.1 GPa[a]
Poisson's ratio	0.33[a]
Density	2780 kg/m^3[a]

[a] www.aerospacemetals.com (accessed on 2nd August 2017)

4.1 Optimization problem

The objective of the optimization problem is to maximize the fundamental natural frequency ω_n[7] of an integrally stiffened aluminum panel, subject to mass and manufacturing constraints. The panel is defined to have a square shape and is manufactured from aluminum alloy 2024-T3, commonly used in aircraft structures (see Table 7 for geometrical and material properties).

The material is intended to be removed via face milling. We impose two manufacturing constraints: the minimum wall thickness is 1 mm, and all stiffeners in a design are defined to have the same size. In addition, all stiffeners are assigned to the same side of the panel (assuming the other side to be wetted by flow). Separately in each design, the stiffener size is scaled so that the total mass of the structure is 8 kg. Therefore, the optimization problem is about finding a suitable trade-off between local and global stiffening. A coarse stiffener layout may not provide adequate support for local plate sections, which become critical for vibration. On the other hand, if the stiffener layout is fine, the stiffener size decreases, and therefore the global oscillation mode involving the entire structure becomes critical.

The fundamental natural frequency ω_n is solved by finite element(FE)-based modal analysis. The analyzes are performed using FE software, Abaqus. The creation of FE models, their execution and post-processing are automated using the Python scripting interface of Abaqus. The boundary conditions are specified to be pinned for all four edges of the panel.

4.2 Reference designs

First, we need to determine the optimal stiffener densities for reference designs with iso- and orthogrids, which maximize the objective function ω_n. We use the number of stiffeners in the x-direction (see the coordinates in Fig. 7) of the panel as a design variable. With orthogrids, the number of stiffeners in y-direction is the same as in x-direction. With isogrids, the numbers of stiffeners in the other two directions are adjusted so that the resulting grid consists of geometries as close to equilateral triangles, or their halves, as possible.

The number of stiffeners is varied with both iso- and orthogrids from 0 to 10. The results are plotted in Fig. 7, where fundamental mode shapes of representative designs are shown. The maximum fundamental natural frequency (ω_n = 143.39 Hz) corresponds to the orthogrid design with five stiffeners in both directions, which we use as the baseline. As a comparison, the highest fundamental natural frequency with an isogrid is 136.93 Hz.

4.3 Topology optimization setup and results

Map L-systems produce a two-dimensional phenotype, which in this application describe the stiffener topology of the plate structure. The plate structure, as well as its boundary conditions, has two perpendicular symmetry axes. We make a priori assumption that the optimal topology is also symmetric at least with respect to one of these axes. Thus, we define the axiom of map L-systems to be a rectangle, covering half of the plate area (see nodes 1–4 and the continuous lines in Fig. 8a), and the second half is generated by mirroring the first half with respect to the vertical symmetry axis (see the dashed lines in Fig. 8a). The boundary edges $\{1, 2\}$, $\{2, 3\}$, $\{3, 4\}$ and their mirrored counterparts are excluded from the stiffener topology. The existence of edge $\{4, 1\}$, laying on the symmetry axis, is decided by an additional, boolean design variable.

Let us now examine the choice of control parameters, the matter at the core of this paper. In Section 3 we showed a systematic process for identifying the optimal parameter combination for a particular problem and we demonstrated this across a set of 'lightweight' test functions. Ideally, we would deploy a similar strategy here too, but, of course, the computational cost of the FE analysis would render this method entirely impractical. Using the same number of CPUs, as in the case of the test functions, would result in a total wall time of 520 days on this class of 'real' problem. We therefore propose the next best strategy: using the conclusions drawn from the Pareto study of the test functions to choose the control parameters of the expensive problem. This is based on the observation that the presence of the L-system as an intermediary level should reduce the problem-specificity of the findings of Section 3; we put forward the correlation analysis presented there (showing similarity of behavior across the test set) as supporting

[7] A structure is resistant to vibration caused by an external excitation with frequency lower than its fundamental natural frequency. Therefore, a high fundamental natural frequency enables vibration-free operation of the structure under a broad range of excitation frequencies.

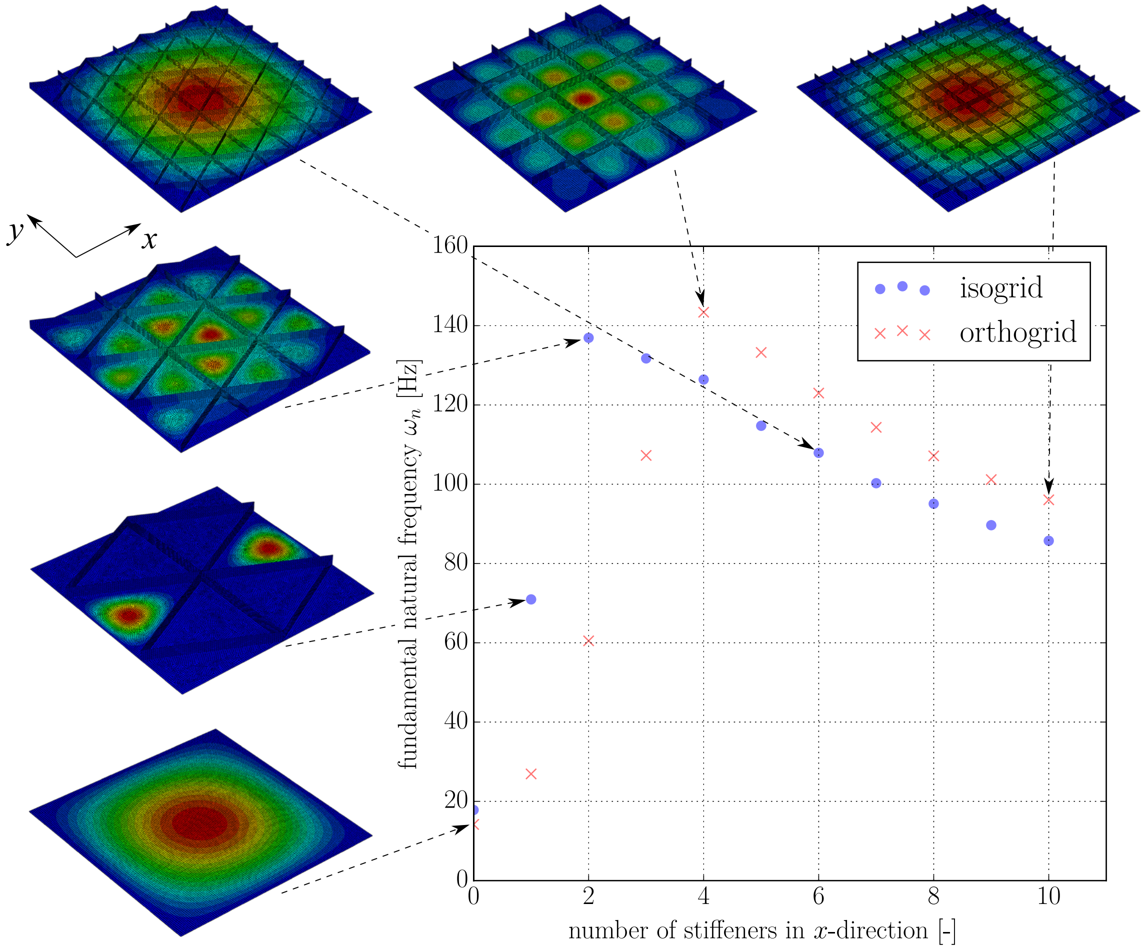


Fig. 7 Reference designs using iso- and orthogrids with variable stiffener densities. The orthogrid with five stiffeners in both directions has the highest fundamental natural frequency of $\omega_\mathrm{n} = 143.39$ Hz

evidence. Additionally, the geometry of this problem is very similar to that of the test cases, even if the objective function itself is not.

Thus, we use here control parameter combination #7 (Table 4), which we found to have the smallest estimate of required objective function evaluations to discover the global optimum. The estimate is based on the average completion rate of Test Cases 1-4, and has a confidence of 95%. In addition, we study the trade-off between the optimized objective function value and

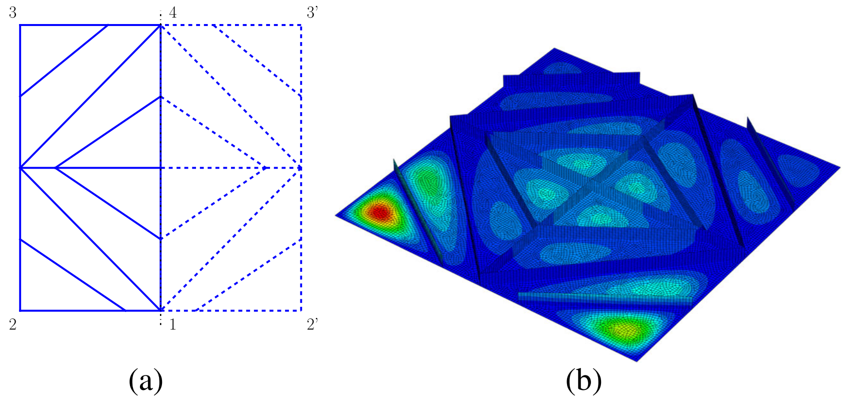


Fig. 8 The best design (run 4) obtained by the map L-systems-based topology optimization. Subfigure **a** shows the optimized map as a two-dimensional graph, where the actual map is drawn with continuous lines and the mirrored side with dashed lines. Subfigure **b** presents the corresponding integrally stiffened panel. The contours illustrate the mode shape of its fundamental natural frequency

Table 8 Comparison of topology optimization (TO) results with parameter combinations #1, #7 and #12 against the reference designs, obtained using iso- and orthogrids

Method	ω_n [Hz]	relative difference [%]	$\bar{Q}$ [10^3]
Isogrid	136.93	−4.51	–
Orthogrid	143.39	0.0	–
TO: #1	147.51 ± 4.61	2.87	14.78
TO: #7	149.32 ± 5.27	4.14	7.18
TO: #12	146.51 ± 7.54	2.17	3.72
TO: best	159.45	11.20	9.60

ω_n and $\bar{Q}$ are the optimized, fundamental natural frequency and the average number of required objective function evaluations, respectively

the number of required objective function evaluations on the application by also running experiments with control parameter combinations #1 and #12. These two combinations represent the end points of the 'practical' Pareto front. The optimization process is repeated 10 times with each of the three combinations.

The average and standard deviation of optimized objective function values, i.e. the fundamental natural frequency ω_n, and the average number of objective function evaluations, $\bar{Q}$, are listed in Table 8. The table also shows the corresponding values for the reference designs, as well as relative differences with respect to the best reference design, i.e. the orthogrid with four stiffeners in x-direction. All three tested parameter combinations yield on average better designs than the orthogrid design.

Based on the results, tested parameter combinations #1, #7 and #12 have a decreasing order of the average number of required objective function evaluations—in the same way as in the statistical experiment. Further, parameter combinations #1 and #7 yield, on average, better designs than parameter combination #12—again in the same way as in the statistical experiment. However, parameter combination #7 yields, on average, better designs than parameter combination #1, which is in contrast to that obtained from the statistical experiment. Due to the high computational cost of the application, and resulting small sample size, we are unable to draw here any statistically significant conclusions. Nevertheless, the results of the application have some similar trends to those of the statistical experiment.

The best design (Fig. 8b), which convergence history is shown in Fig. 9, was obtained using the parameter combination #7. The design has 11.20% higher fundamental natural frequency than the orthogrid design. It consists of two radial lines of stiffener, laying at the two perpendicular symmetry axes, and three circumferential stages of stiffeners. While mirroring is applied with respect to the vertical symmetry axis, the design is also nearly-symmetric with respect to the horizontal symmetry axis. The corresponding map L-system and its additional variables are

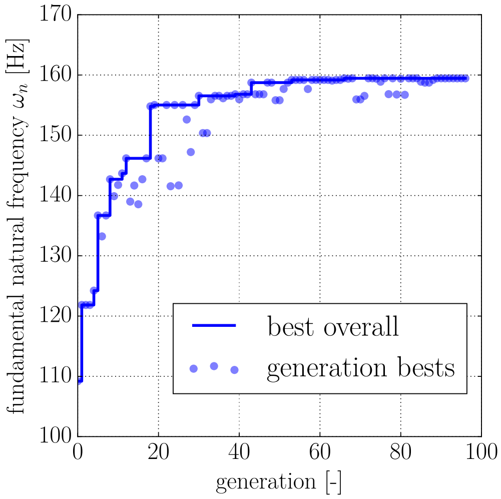


Fig. 9 Convergence history of optimization run 4. The best design is discovered at the 66th generation

$$
\begin{array}{ll}
\text{Axiom:} & \omega_0 = CCAC \\
\text{Rules:} & P_1 : A \rightarrow [+\overrightarrow{D}][+B]C[+\overleftarrow{D}]A \\
 & P_2 : B \rightarrow [-B][+D][-D]C[-\overrightarrow{D}][+\overrightarrow{B}] \\
 & P_3 : C \rightarrow [+D]B[-D]B[-\overleftarrow{C}] \\
 & P_4 : D \rightarrow [-\overrightarrow{B}]DB[+C]CD \\
\text{Additional variables:} & f_\mathrm{a} = 0.43263 \\
 & n = 3
\end{array}
\tag{13}
$$

It is also worth mentioning that the second and third lowest natural frequencies of the design are only 0.14% and 0.82%, respectively, higher than the fundamental natural frequency. As a comparison, the corresponding values of the orthogrid design are both 16.62% higher than the fundamental natural frequency.

Let us review the obtained designs from the five test cases (Figs. 2 and 3) and the application (Fig. 8). Although the designs are significantly different from each other, depending on their objective function, they are still obtained using the same type of generative encoding. The used methodology does not require a priori definitions of mesh resolution or the type or density of the ground structure.

Despite being more concise than an equivalent direct encoding method, the number of design variables in the L-systems-based parameterization is still relatively large, but well within the capabilities of a GA, as illustrated, for example, by the success of the nearly 400-variable study of Pedro and Kobayashi (2011).[8] In this particular case

[8] An estimate based on the fact that their design space consists of eight rewriting rules, which each consist of eight tokens.

we have 126 design variables, which is a very concise description of a vast design space: as shown in Table 3, the number of different L-systems in the design space is of the order of $3 \cdot 10^{36}$, excluding any variation in the scalar variable f_a. The effectiveness of the GA is also underlined by the fact that we were able to find global optima in four out of five test cases, and it yielded better designs than the conventional engineering process in the application.

5 Conclusions

The main goal of this study was to examine the effects of genetic control parameters on the performance of the map L-systems-based topology optimization method. A total of 432 control parameter combinations were tested on five test cases, with known global optima. The results show that carefully chosen control parameter combination can significantly increase the performance of the map L-systems-based topology optimization. The Pareto front of best performing parameter combinations is reported. These parameter combinations are recommended starting points for a designer using the map L-systems-based topology optimization, with a numerical representation similar to the one described by Pedro and Kobayashi (2011).

The pairwise comparisons of parameter combination ranks in between the test cases show strong correlation (the Spearman's rank correlation coefficient ρ ranges from 0.645 to 0.979), which indicates that a parameter combination, performing well on one test case, is also likely perform well on another test case. In addition, we found strong correlation (ρ ranges from 0.800 to 0.988) between the parameter combination ranks obtained using different numbers of rewriting rules on Test Case 1. The result is an indication that the guidelines we give in this paper are applicable also to studies with a number of rewriting rules different to what we use here.

Finally, we demonstrated the application of these findings to an engineering design problem, that of an integrally stiffened aluminum panel; specifically, we showed how to establish the appropriate computational effort and how to select the most suitable combination of control parameters. The best obtained design was 11.2% better, in terms of its fundamental natural frequency, than the baseline design obtained through conventional means.

Acknowledgments Theauthors are grateful to the Finnish Cultural Foundation, and the Engineering and Physical Sciences Research Council (UK) for sponsoring the work. In addition, the authors acknowledge the use of the IRIDIS High Performance Computing Facility at the University of Southampton in the completion of this work.

The electronic supplementary material is also openly available from the University of Southampton repository at https://doi.org/10.5258/SOTON/D0431.

Appendix A: Numerical representation of map L-systems

This appendix describes the encoding of map L-systems into a numerical representation (suitable for genetic algorithms), used in this work. The encoding is similar to that described by Pedro and Kobayashi (2011). Parts of the description are taken from reference Ikonen and Sóbester (2017).

The axiom, rewriting rules and additional variables are encoded sequentially into a vector $\mathbf{x}$ of real numbers, with $x_i \in [0, 1] \forall i$, as

$$\mathbf{x} = [\,\underbrace{x_{\mathrm{a},1}\ x_{\mathrm{a},2}\ \dots\ x_{\mathrm{a},N_\mathrm{a}}}_{\text{Axiom } \omega_0}\ \underbrace{P_1\ P_2\ \dots\ P_{N_\mathrm{P}}}_{\text{Rewriting rules} P_j}\ \underbrace{x_1\ x_2\ \dots\ x_{N_\mathrm{v}}}_{\text{Additional variables}}\,]. \tag{14}$$

Each letter of the axiom, having a total of N_a letters, is represented as a real number $x_{\mathrm{a},i}$. The interval of the real number is divided into equally sized segments representing the letters in an alphabet Σ. For example, if the alphabet is $\Sigma = \{A, B, C\}$, a real number $x_{\mathrm{a},i}$ in the axiom is assigned the following segments: $A \equiv [0, \frac{1}{3}]$, $B \equiv [\frac{1}{3}, \frac{2}{3}]$, $C \equiv [\frac{2}{3}, 1]$.

The total number of rewriting rules, N_P, is equal to the length of the alphabet. Each rewriting rule P_j is encoded into N_r sets of real numbers, called *tokens*, as

$$P_j = \left[\, \beta_{j,1}\ \beta_{j,2}\ \dots\ \beta_{j,N_\mathrm{r}} \,\right]. \tag{15}$$

A token is a part of the right-hand side of a rewriting rule, and may appear as a neutral letter, a marker or an empty token. Examples of the first two instances are A and $[-\overleftarrow{B}]$, respectively. The number of encoded tokens, N_r, defines the maximum length of the rewriting rule. The kth token of the jth rewriting rule is encoded into a set of five real numbers as

$$\beta_{j,k} = \left[\, x_1\ x_2\ x_3\ x_4\ x_5 \,\right], \tag{16}$$

where the real numbers encode the token as follows:

1. Existence of the token: an empty token if x_1 in $[0, p_\mathrm{empty}]$, else token is non-empty. If the token is empty, the real numbers $x_2, \dots x_5$ are ignored.
2. Edge letter: A if x_2 in $[0, \frac{1}{N_\mathrm{a}}]$, else B if x_2 in $[\frac{1}{N_\mathrm{a}}, \frac{2}{N_\mathrm{a}}]$, else C if x_2 in $[\frac{2}{N_\mathrm{a}}, \frac{3}{N_\mathrm{a}}], \dots$

3. Marker orientation[9] '→' if x_3 in $[0, \frac{1}{3}]$, else n (neutral) if x_5 in $[\frac{1}{3}, \frac{2}{3}]$, else '←'.
4. Marker: the token is a marker if x_4 in $[0, p_{\text{marker}}]$, else the token is a neutral letter.
5. Marker side: the side is '+' if x_5 in $[0, \frac{1}{2}]$, else the side is '−'.

Pedro and Kobayashi (2011) define an additional, sixth, real number to vary a specific property of the edge (e.g. the thickness). The edge property is redundant in our test cases and application, and therefore omitted.

Finally, N_{v} additional variables are encoded into the vector $\mathbf{x}$. The additional variables always contain the age n of the system, i.e. the number of applied developmental stages, which is an integer variable with lower and upper limits. Each integer value in the range is assigned an equal interval of the real number. The additional variables can be amended by additional requirements for new cycles. These requirements can define, for example, the minimum angle between two edges belonging to a cycle, the minimum fraction for the area of an offspring cycle in comparison to the parent cycle, or the minimum fraction for the shortest edge in comparison to the longest edge in a cycle. These variables are scaled to the encoding interval of [0, 1].

As a summary, the design variable vector $\mathbf{x}$ has a total length of

$$N_{\text{total}} = N_{\text{a}} + 5N_{\text{r}}N_{\text{P}} + N_{\text{v}}. \tag{17}$$

References

Allison JT, Khetan A, Lohan DJ (2013) Managing variable-dimension structural optimization problems using generative algorithms. In: Proceedings of the 10th world congress on structural and multidisciplinary optimization (WCSMO), Orlando

Baker JE (1987) Reducing bias and inefficiency in the selection algorithm. In: Proceedings of the second international conference on genetic algorithms, pp 14–21

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

Blickle T, Thiele L (1995) A comparison of selection schemes used in genetic algorithms

Chomsky N (1956) Three models for the description of language. IRE Trans Inf Theory 2(3):113–124

Corder GW, Foreman DI (2014) Nonparametric statistics: a step-by-step approach. Wiley, New York

Deaton JD, Grandhi RV (2014) A survey of structural and multidisciplinary continuum topology optimization: post 2000. Struct Multidiscip Optim 49(1):1–38

El-Soudani SM (2006) Methods of making integrally stiffened axial load carrying skin panels for primary aircraft structure and fuel tank structures. US Patent 7,093,470

Hornby GS, Pollack JB (2001) The advantages of generative grammatical encodings for physical design. In: Proceedings of the 2001 congress on evolutionary computation, vol 1. IEEE, pp 600–607

Hornby GS, Pollack JB (2002) Creating high-level components with a generative representation for body-brain evolution. Artif Life 8(3):223–246

Huang X, Xie M (2010) Evolutionary topology optimization of continuum structures: methods and applications. Wiley, New York

Huybrechts SM, Meink TE, Wegner PM, Ganley JM (2002) Manufacturing theory for advanced grid stiffened structures. Compos A: Appl Sci Manuf 33(2):155–161

Ikonen TJ, Sóbester A (2017) Two variations to the map L-systems-based topology optimization method. In: Proceedings of the 17th AIAA aviation technology, integration, and operations conference, Denver

Kobayashi MH (2010) On a biologically inspired topology optimization method. Commun Nonlinear Sci Numer Simul 15(3):787–802

Kobayashi MH, LeBon A, Pedro HTC, Kolonay RM, Reich GW (2010) On a cellular division model for multi-disciplinary optimization. In: 51st AIAA/ASME/ASCE/AHS/ASC structures, structural dynamics, and materials conference, p 2989

Kolonay RM, Kobayashi MH (2010) Topology, shape, and sizing optimization of aircraft lifting surfaces using a cellular division method. In: Proceedings of the 13th AIAA/ISSMO multidisciplinary analysis optimization conference

Kolonay RM, Kobayashi MH (2015) Optimization of aircraft lifting surfaces using a cellular division method. J Aircr 52(6):2051–2063

Lindenmayer A (1968a) Mathematical models for cellular interactions in development i. filaments with one-sided inputs. J Theor Biol 18(3):280–299

Lindenmayer A (1968b) Mathematical models for cellular interactions in development ii. simple and branching filaments with two-sided inputs. J Theor Biol 18(3):300–315

Mitchell M (1998) An introduction to genetic algorithms. MIT Press, Cambridge

Nakamura A, Lindenmayer A, Aizawa K (1986) Some systems for map generation. In: The Book of L. Springer, Berlin, pp 323–332

Pedro HTC, Kobayashi MH (2011) On a cellular division method for topology optimization. Int J Numer Methods Eng 88(11):1175–1197

Perone CS (2009) Pyevolve: a python open-source framework for genetic algorithms. ACM SIGEVOlution 4(1):12–20

Prusinkiewicz P, Lindenmayer A (2012) The algorithmic beauty of plants. Springer Science & Business Media, Berlin

Rieffel J, Valero-Cuevas F, Lipson H (2009) Automated discovery and optimization of large irregular tensegrity structures. Comput Struct 87(5):368–379

Rozenberg G, Salomaa A (2012) Lindenmayer systems: impacts on theoretical computer science, computer graphics, and developmental biology. Springer Science & Business Media, Berlin

Sabbatini E, Revel GM, Kobayashi MH (2015) Vibration reduction using biologically inspired topology optimization method: optimal stiffeners distribution on an acoustically excited plate. J Vib Control 21(7):1398–1412

Simpson TW, D'souza BS (2004) Assessing variable levels of platform commonality within a product family using a multiobjective genetic algorithm. Concurr Eng 12(2):119–129

Stanford B, Beran P, Kobayashi MH (2012) Aeroelastic optimization of flapping wing venation: a cellular division approach. AIAA J 50(4):938–951

Stanford B, Beran P, Kobayashi MH (2013) Simultaneous topology optimization of membrane wings and their compliant flapping mechanisms. AIAA J 51(6):1431–1441

Stanley KO, Miikkulainen R (2003) A taxonomy for artificial embryogeny. Artif Life 9(2):93–130

[9] In the encoding by Pedro and Kobayashi (2011) this element also defines the edge direction (in the case of a neutral letter). For simplicity, we define all edges to have a uniform direction.