

University of Southampton Research Repository

Copyright © and Moral Rights for this thesis and, where applicable, any accompanying data are retained by the author and/or other copyright owners. A copy can be downloaded for personal non-commercial research or study, without prior permission or charge. This thesis and the accompanying data cannot be reproduced or quoted extensively from without first obtaining permission in writing from the copyright holder/s. The content of the thesis and accompanying research data (where applicable) must not be changed in any way or sold commercially in any format or medium without the formal permission of the copyright holder/s.

When referring to this thesis and any accompanying data, full bibliographic details must be given, e.g.

Thesis: Author (Year of Submission) "Full thesis title", University of Southampton, name of the University Faculty or School or Department, PhD Thesis, pagination.

Data: Author (Year) Title. URI [dataset]

UNIVERSITY OF SOUTHAMPTON

FACULTY OF ENGINEERING AND THE ENVIRONMENT

Aeronautics, Astronautics and Computational Engineering

**Evolutionary Topology Optimization via Direct and Generative
Encodings: Applications to Aerospace and Heat Transfer Engineering**

by

Teemu Johannes Ikonen

Thesis for the degree of Doctor of Philosophy

September 21, 2018

UNIVERSITY OF SOUTHAMPTON

ABSTRACT

FACULTY OF ENGINEERING AND THE ENVIRONMENT

Aeronautics, Astronautics and Computational Engineering

Doctor of Philosophy

EVOLUTIONARY TOPOLOGY OPTIMIZATION VIA DIRECT AND
GENERATIVE ENCODINGS: APPLICATIONS TO AEROSPACE AND HEAT
TRANSFER ENGINEERING

by Teemu Johannes Ikonen

Evolutionary algorithms are global search methods that are well-suited for ‘black-box’ type objective functions and multi-objective optimization. However, as search methods in topology optimization, they have gained only limited acceptance, mainly due to their poor efficiency; they tend to require more objective function evaluations than gradient-based methods. Motivated by their benefits, the first aim of this work is to improve the performance, i.e. effectiveness and efficiency, of evolutionary topology optimization. We parameterize the design domains using both the ground structure approach (direct encoding) and L-systems-based methods (generative encoding). We investigate the use of two interpretation formalisms of L-systems, i.e. map L-systems and the turtle interpretation. In terms of improving the performance, the main contribution of this work is a statistical analysis of the effects of over 400 genetic control parameter combinations on the performance of the map L-systems-based method, which results we report as a Pareto front in the space of effectiveness and efficiency. The second aim of this work is to identify engineering applications to which L-systems-based methods are particularly suitable. We studied three applications, which are related to aerospace and heat transfer engineering. We found that the method with the turtle interpretation is well-suited to topology optimization of a heat conductor due to its natural tendency to produce bifurcating tree-structures. We show that the method is more effective in 10 out of 12 tested optimization problems and is two orders of magnitude more efficient on all 12 problems than a representative direct encoding method. In addition, our results indicate that the method is more effective than the well-established SIMP method (Solid Isotropic Material with Penalization) in optimization problems where the product of volume fraction and the ratio of high and low conductive material is less or equal to 1.

Declaration of Authorship

I, **Teemu Johannes Ikonen**, declare that the thesis entitled *Evolutionary Topology Optimization via Direct and Generative Encodings: Applications to Aerospace and Heat Transfer Engineering* and the work presented in the thesis are both my own, and have been generated by me as the result of my own original research. I confirm that:

- this work was done wholly or mainly while in candidature for a research degree at this University;
- where any part of this thesis has previously been submitted for a degree or any other qualification at this University or any other institution, this has been clearly stated;
- where I have consulted the published work of others, this is always clearly attributed;
- where I have quoted from the work of others, the source is always given. With the exception of such quotations, this thesis is entirely my own work;
- I have acknowledged all main sources of help;
- where the thesis is based on work done by myself jointly with others, I have made clear exactly what was done by others and what I have contributed myself;
- parts of this work have been published as: (**Ikonen and Söbester, 2016, 2017, 2018, Ikonen et al., 2018**)

Signed:.. 

Date:.. 21st of September 2018

Acknowledgements

I am very grateful to my primary supervisor András Sóbester for enabling me to conduct this work, as well as for providing me excellent guidance and advice during the work. Further, I would like to thank Andy Keane and James Scanlan for their insight to aircraft design and evolutionary algorithms, and Atul Bhaskar and Neil Bressloff for their valuable comments on how to improve the work.

I wish to thank Gilles Marck for the fruitful collaboration during the heat transfer project. Your fearless and creative attitude towards new challenges has been very inspiring. I also wish to thank Willem Eerland, Chris Chrispin and Chris Paulson for their help and ideas with Python programming.

Finally, I would like to thank my parents for always support me.

Contents

Acknowledgements	vi
Nomenclature	xix
1 Introduction	1
1.1 Aims of the research	3
1.2 Structure of the thesis	4
2 Topology optimization	5
2.1 Introduction to optimization	5
2.2 Design space parameterization	7
2.2.1 Direct encodings	7
2.2.2 Geometric encodings	7
2.2.3 Generative encodings	8
2.3 Genetic algorithms (GAs)	10
2.3.1 Encoding	11
2.3.1.1 Multi-dimensional genotypes in direct encoding	11
2.3.2 Genetic operators	12
2.3.3 GAs with constraints	14
2.3.4 Multi-objective GAs	16
2.4 Ground structure approach	18
2.5 Evolutionary structural optimization (ESO)	20
2.5.1 Sensitivity numbers	21
2.5.2 Bi-directional ESO (BESO)	23
2.6 Other topology optimization methods	24
2.6.1 Solid Isotropic Material with Penalization (SIMP)	25
2.6.2 Level-set method (LSM)	26
2.7 Lindenmayer systems (L-systems)	28
2.7.1 Turtle interpretation	29
2.7.1.1 Turtle interpretation of parametric symbols	30
2.7.2 Map L-systems	32
2.7.3 Optimization via a GA	34
2.7.4 Applying L-systems to topology optimization	36
2.8 Selected areas of application	38
2.8.1 Aircraft wing structures	39
2.8.1.1 Design space parameterization	39
2.8.1.2 Optimization methods	43

2.8.1.3	Buckling constraints	44
2.8.1.4	Parametric geometries	45
2.8.2	Conductive heat transfer systems	47
2.8.3	Integrally stiffened panels: natural frequency maximization	49
2.9	Conclusions	53
3	The ground structure approach to structural topology optimization	57
3.1	Procedure	57
3.2	Geometry generation	59
3.3	Ground structures and the component hierarchy	59
3.4	GA-based optimization method	61
3.5	(B)ESO-based optimization method	64
3.6	Constraint evaluation (FE analysis)	66
3.7	Application I: Topology optimization of a sUAV wing	68
3.7.1	Description of the design problems	68
3.7.2	Conventional design methods	71
3.7.3	Ground structures for the application	72
3.7.4	Results and discussion	73
3.7.4.1	Mesh density verification	74
3.7.4.2	The use of penalty function in GA-based optimization	75
3.7.4.3	Population size in the GA-based method	77
3.7.4.4	Crossover types in the GA-based method	79
3.7.4.5	(B)ESO-based optimization parameters	83
3.7.4.6	Choosing the parameters	86
3.7.4.7	Quadrilateral (GS1) and diagonal (GS2) ground structures	87
3.7.4.8	Hexagonal ground structures (GS3)	89
3.7.4.9	Comparison to conventional design methods	92
3.8	Conclusions	94
4	Map L-systems-based method: statistical experiment	95
4.1	Defining the design space	96
4.2	Introductory example	97
4.2.1	Test Case 1	97
4.2.2	Results	99
4.3	Experimental plan	100
4.4	Remaining test cases	103
4.4.1	Test Case 2	103
4.4.2	Test Case 3	103
4.4.3	Test Case 4	104
4.4.4	Test Case 5	105
4.5	Results and discussion	105
4.6	Conclusions	112
5	Map L-systems-based method: two proposed improvements	114
5.1	Linearization of the age variable	114
5.1.1	Demonstration on Test Case 1	117
5.1.2	Statistical experiment	120

5.2	Implementation of the component hierarchy	124
5.3	Application I continued: Topology optimization of a sUAV wing	125
5.3.1	Defining the design space	126
5.3.2	Results and discussion	127
5.4	Conclusions	131
6	Application II: Heat conductor	132
6.1	Optimization problem	133
6.2	Methods	135
6.2.1	Ground structure approach	136
6.2.2	L-systems-based method	139
6.2.3	Mapping	142
6.2.4	Finite Volume Method (FVM)	144
6.3	Results	146
6.3.1	Control parameters in the ground structure approach	147
6.3.2	Ground structure type and density	150
6.3.3	Ground structure approach versus L-systems-based method	152
6.3.4	Boichot's problem parameters	155
6.3.5	Variation of the L-systems-based method parameters	156
6.3.5.1	Control parameters	157
6.3.5.2	Number of encoded rewriting rules	158
6.3.6	Benchmarking on single-objective optimization problems	160
6.3.7	Multi-objective optimization	167
6.4	Conclusions	169
7	Application III: Integrally stiffened panel	171
7.1	Optimization problem	171
7.2	Methods	173
7.2.1	Ground structure approach	173
7.2.2	Map L-systems-based method	175
7.2.3	Scaling	176
7.3	Reference designs	176
7.4	Mesh density verification	178
7.5	Results	179
7.5.1	Ground structure type and density	179
7.5.2	Control parameters in the map L-systems-based method	182
7.5.3	Benchmarking	185
7.6	Conclusions	186
8	Conclusions	188
8.1	Improving the performance	188
8.2	Suitability of generative encodings	190
8.3	On patterns of similar structural members	191
8.4	Publications	192
8.5	Future work	192
8.5.1	Exploring the search algorithms	192
8.5.2	More complex heat transfer problems	192

8.5.3 Varying the optimization problem setups of integrally stiffened panels	193
A Details of the statistical experiment in Section 5.1.2	194
B Optimal map L-systems for Test Cases 2-5	197
C Collaboration with Dr. Gilles Marck	199
D Detailed description of the projection method used in Section 6.2.3	200
E Distributions of optimized objective function values in Section 6.3.6	202
F Optimized designs for optimization problems 3-5 and 9-11 in Section 6.3.6	205
G List of publications	208
Bibliography	209

List of Figures

1.1	Designs consisting of similar structural members.	2
2.1	Construction of a phenotype in stages using a cell chemistry approach, based on motile polarized cells (Steiner et al., 2009). The last figure represents discretization into a three-dimensional grid.	9
2.2	An illustration of the CPPN method to encode a phenotype, adopted from the paper by Cheney et al. (2014).	10
2.3	Volume-constrained stiffness maximization of a cantilever structure using BESO (Huang and Xie, 2010a). The initial guess (a) and the optimized structure (b) are presented.	24
2.4	Topology optimization of a short cantilever beam using the SIMP method (Sigmund, 2001).	26
2.5	The development of the structural topology in a meshless level-set optimization process (Luo et al., 2012). The structural topology is plotted at generations 1 (Subfigure a), 25 (b), 75 (c), 150 (d), 300 (e) and 481 (f).	27
2.6	Level-set functions corresponding to the structural topologies presented in Figure 2.5 (Luo et al., 2012).	27
2.7	Visualization of a growing plant, interpreted from the L-system defined in Equation 2.19. The variable n is ordinal of the developmental stage. Thus, $n = 0$ corresponds to the axiom ω_0 , and $n = 1 \dots 3$ to the three subsequent developmental stages of the plant.	30
2.8	Visualization of the parametric L-system defined in Equation 2.22.	31
2.9	Visualization of a cellular division method. The graph shows the axiom ($n = 0$) and first four developmental stages ($n = 1 \dots 4$) of a map L-system (redrawn from the text book by Prusinkiewicz and Lindenmayer (2012)).	34
2.10	Stress-constrained mass minimization of a cantilever structure with the map L-system-based parameterization (Pedro and Kobayashi, 2011). The design space is a rectangular area (a), where the lower- and uppermost points on the left-hand side boundary are fixed and a load is applied on the center point on the right-hand side boundary. The optimized structure (b) has a volume that is 26% of the total the design space volume.	37
2.11	Continuum and discrete material definitions in topology optimization studies on aircraft wing structures.	40
2.12	Optimized design, minimizing the compliance of the NASA Common Research Model wing, obtained by the SIMP method (Aage et al., 2017).	41
2.13	Integrally stiffened panels	50
2.14	Example of designs with optimized fundamental natural frequencies obtained using the SIMP method (a) and BESO heuristics (b).	52

3.1	The topology optimization procedure via the ground structure approach. .	58
3.2	Quadrilateral (a), quadrilateral with diagonals (b) and hexagonal (c) ground structures.	60
3.3	The component hierarchy.	60
3.4	An extended component hierarchy.	61
3.5	Example of crossover between two-dimensional strings representing wing designs.	62
3.6	Example splices of crossovers on one- and two-dimensional strings, where X represents an element, and blue and yellow colors indicate genes inherited from parents 1 and 2, respectively.	63
3.7	The pressure load f_p and inertia load f_g applied to the wing.	67
3.8	Modal shapes of the first critical buckling load for wing designs with traditional methods.	70
3.9	Two examples of conventional internal structures. The internal structures have been designed manually using conventional spars, ribs and stiffeners.	72
3.10	Three ground structures used in the study. The member density of Ground structure 3 is varied (see Figure 3.11).	73
3.11	Four member densities for ground structure 3. The ground structures have a total of 141 (Subfigure a), 186 (b), 238 (c) and 295 members (d).	73
3.12	Mesh density verification of the full GS1 and an optimized ground structure.	75
3.13	Representation of individuals in a GA-based optimization run. Subfigure (a) presents the normalized critical buckling load of individuals as a function of their generation, while subfigure (b) plots the individuals of selected generations based on their critical buckling load and structural mass.	76
3.14	Contour plot of the score of all feasible individuals during the optimization process. Infeasible individuals are indicated by grey.	77
3.15	Convergence histories of experiments with different population sizes N_{pop}	78
3.16	Comparison of the two-point (one-dimensional encoding), random walk (two-dimensional encoding) and distributed crossover.	80
3.17	Statistical results of the required generations to converge (a) and the optimized mass (b) on the tree experimented crossovers, and the corresponding normal distribution. Optimization runs with each crossover are repeated 10 times.	82
3.18	Variation of the internal structure mass (a) and the critical buckling load (b) during (B)ESO-based optimization runs on GS1.	85
3.19	Evolution of the internal structure using the GA-, ESO- and BESO-based optimization methods on GS1.	88
3.20	Optimized structures from optimization runs initiated from the diagonal ground structure (GS2). Results with GA, ESO and BESO are presented.	89
3.21	Optimized structures from optimization runs initiated from the four different ground structure densities of the hexagonal ground structure (GS3).	90
3.22	Optimized internal structure masses from GS1, GS2 and GS3 with four ground structure densities.	91
3.23	Comparison of the lightest obtained design from topology optimization against the conventional design methods.	93
4.1	A map L-system producing a square grid.	98

4.2	Representative optimization runs yielding the global optimum and a sub-optimal final result.	99
4.3	Example global optima in Test Cases 2-4, and the design with the lowest objective function value in Test Case 5. The map L-systems of these phenotypes are listed in Appendix B.	103
4.4	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. Each of Subfigures (a)-(f) visualizes the effect of one control parameter on the performance of the algorithm. The dashed line represents the Pareto front between minimum average number of function evaluations and maximum completion rate.	106
4.5	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.	109
5.1	The development of the map L-system presented in Figure 2.9 using the number of cell divisions, n_{lin} , as the age variable.	115
5.2	The number of cells in the map L-system described in in Equation 2.24 as a function of the two alternative design variables: baseline age n (circles) and linear age n_{lin} (crosses). The baseline age is the ordinal of the developmental stage, whereas the linear age defines the number of cell divisions.	116
5.3	The best individuals of a GA optimization process, where the objective is to match the regular grid of 16 cells (Test Case 1). Representative individuals during the optimization process are shown.	118
5.4	First 15 cell divisions of the map L-system described in equation 5.1. The cell divisions are performed at the following developmental stages: 1st (developmental stage): $n_{lin} = 1$, 2nd: $n_{lin} = 2$, 3rd: $n_{lin} = 3$, 4th: $n_{lin} = 4 \dots 6$, 5th: $n_{lin} = 7 \dots 10$, 6th: $n_{lin} = 11 \dots 15$	119
5.5	The completion rate p_c and average number of objective function evaluations, $\bar{Q}$, on the five test cases (TC) using algorithms with different age variables (a). The algorithm with the linear age variable is tested with and without the local search (LS). No global optimum was found for TC5; thus, we also report its average optimized objective function value $\bar{f}_{min}$ (b).	122
5.6	An example of mapping a map L-system with component hierarchy into a wing internal structure.	125
5.7	Distributions of optimized mass m_{IS} (a) and required objective function evaluations, Q (b), using the map L-systems-based method with baseline and linear age parameterizations. As a reference, we have included the corresponding results obtained using the ground structure approach with GS1.	128
5.8	Optimized designs obtained using the map L-systems-based method.	129
5.9	The evolution process of design shown in Figure 5.8(b).	130

6.1	The optimization problem of optimally distributing a limited amount of high conductive material (domain Ω_p), to minimize the average, or maximum, temperature in domain $\Omega_p \cup \Omega_0$	134
6.2	The procedure of the optimization process.	135
6.3	An example of a bifurcating tree-structure, represented using nodes 1-6 (in <i>black</i>) and elements 1-5 (in <i>grey</i>).	136
6.4	Two types of ground structures providing candidate paths for the high conductive material in the half-domain (the upper half of Figure 6.1). The numbering in subfigure (a) indicates the order, in which the structural members are encoded into the genotype.	138
6.5	Projection of two example isosceles trapezoids, without (a) and with (b) additional isosceles trapezoids at their ends, into a Cartesian design grid.	142
6.6	Visualization of the buffer domain Ψ located next to the south and west boundaries of the upper half of domain Ω . (This figure is prepared by Dr. Gilles Marck.)	143
6.7	Staggered finite volume grid. (This figure is prepared by Dr. Gilles Marck.)	145
6.8	An illustration of scaling (upper part), projection (lower part) and evaluating the temperature distribution (lower part) of the slightly modified version of the parametric L-system, plotted earlier in Figure 2.8(d). We have modified it by introducing the trapezoidal variables (see Section 6.2.2)	146
6.9	Convergence histories and distributions of optimized average temperature $\bar{T}$ of the ground structure approach. Optimization runs with each of the three studies control parameter combinations are repeated 30 times.	148
6.10	Distributions of the optimized average temperature $\bar{T}$ and the number of objective function evaluations Q with the three tested control parameters combinations. The data sets contain 30 repeats, and their averages are indicated by vertical lines.	149
6.11	Distributions of optimized average temperature $\bar{T}$ and number of objective function evaluations Q of optimization runs initiated from quadrilateral (with diagonals) and hexagonal ground structures with various densities. The data sets contain 30 repeats, and their averages are indicated by vertical lines.	151
6.12	Best designs obtained using different densities of quadrilateral (with diagonals) and hexagonal grounds structures.	151
6.13	Convergence histories and distributions of optimized average temperature $\bar{T}$ of 30 optimization runs with the L-systems-based method.	153
6.14	Comparison of the optimized average temperature $\bar{T}$ and the number of objective function evaluations Q between the ground structure approach and the L-systems-based method.	153
6.15	The best design obtained using the L-systems-based method. The range of the contour map is the same as in Figure 6.12.	154
6.16	Trade-off study of the control parameter combinations on the optimized average temperature $\bar{T}$ and the number of objective function evaluations Q	158
6.17	Trade-off study of the number of encoded rewriting rules, N_P , on the optimized average temperature $\bar{T}$ and the number of objective function evaluations Q	159

6.18	Benchmarking of the L-systems-based method against the direct encoding (Boichot and Fan, 2016) and the SIMP method (Marck et al., 2012) for problems #1-6. $\bar{T}_{\text{SIMP}}$ is the average temperature optimized by the SIMP method.	160
6.19	Comparison of designs obtained for three representative problems #1, #2 and #6.	162
6.20	Benchmarking of the L-systems-based method against the direct encoding method (Boichot and Fan, 2016) for problems #7-12. $T_{\text{max,DE}}$ is the maximum temperature optimized by the direct encoding method.	165
6.21	Comparison of designs obtained for three representative problems #7, #8 and #12.	167
6.22	Results of two multi-objective optimization runs, evolved via NSGA-II. The figures on the sides show representative designs from the Pareto front, indicated by the dashed line.	168
7.1	Rectangular ground structure with diagonals having a density of 10×5	174
7.2	Hexagonal ground structure having a density of 10×5	175
7.3	The third developmental stage of the map L-system in Equation 2.24 mapped as a stiffener layout.	176
7.4	Reference designs using iso- and orthogrids with variable stiffener densities. The orthogrid with four stiffeners in both directions has the highest fundamental natural frequency of $\omega_f = 143.39$ Hz. The contour plots illustrate fundamental mode shapes of representative designs.	177
7.5	Mesh density verification of the baseline design, i.e. the orthogrid with five seeds a side.	178
7.6	Distributions of the optimized fundamental natural frequency ω_f and the number of objective function evaluations Q from optimization runs with various ground structure types and densities. The vertical lines indicate the average values of the quantities.	180
7.7	The best obtained designs using the ground structure approach. The contours indicate the fundamental mode shape.	181
7.8	Hexagonal stiffener layout with a variable stiffener density. The layout with the highest fundamental natural frequency has five cells in the x -direction ($\omega_f = 137.61$ Hz).	182
7.9	Distributions of the optimized fundamental natural frequency ω_f and the number of objective function evaluations Q from optimization runs using the map L-systems-based method with control parameter combinations #1, #4, #7 and #12. The vertical lines indicate the average values of the quantities.	183
7.10	Representative designs obtained using the map L-systems-based method.	184
7.11	The convergence history and representative designs of the optimization run yielding the best design in Figure 7.10(a).	185
D.1	Projection of an element i , spanning between nodes $P_{i,0}$ and $P_{i,0}$, onto a Cartesian grid.	201
E.1	Distributions of the optimized average temperature $\bar{T}$ in optimization problems #1-6	203
E.2	Distributions of the optimized maximum temperature T_{max} in optimization problems #7-12	204

F.1	Comparison of obtained designs for optimization problems #3, #4 and #5. The unit of the contour map is K.	206
F.2	Comparison of obtained designs for optimization problems #9, #10 and #11. The unit of the contour map is K.	207

List of Tables

3.1	Material and manufacturing properties of 3D printed nylon ¹	68
3.2	Geometric definition of the sUAV wing.	69
3.3	Load properties of the sUAV.	69
3.4	Numerical data of the experiment shown in Figure 3.15.	78
3.5	Mean and sample standard deviations (SD) of the required generations to convergence and the optimized mass with the tree tested crossovers. Each of the algorithms using these crossovers is tested 10 times.	80
3.6	The family of statistical tests ($j = 1, 2, \dots, 6$) ranked in decreasing order of significance. In each test, Crossover 1 has on average of the obtained results a better performance than Crossover 2. Based on the results, the null hypothesis H_0 is rejected in tests $j = 1, 2, \dots, 4$ and accepted in tests $j = 5, 6$	81
3.7	Parameter combinations of the executed (B)ESO-based optimization runs.	83
3.8	Optimized masses of quadrilateral (GS1) and diagonal (GS2) ground structures using GA-, ESO- and BESO-based methods.	89
3.9	Optimized masses of the hexagonal ground structures (GS3) with different densities using GA-, ESO- and BESO-based methods.	91
4.1	Definition of the L-system design space. Minimum area fraction f_a and age n are additional variables.	97
4.2	Survey of control parameters in L-systems-based topology optimization studies found in the literature.	101
4.3	Control parameter values of the statistical experiment. All parameter combinations, totalling 432, are tested separately.	102
4.4	The parameter combinations lying on the Pareto front in Figure 4.5. The combinations are listed in increasing order of the average rank. An extended version of the table, including all tested parameter combinations, is openly available at http://dx.doi.org/10.5258/SOTON/D0431	110
4.5	Pairwise Spearman's rank correlation coefficients ρ_s between the test cases. The ranks are ordered based on the average minimized objective function value (a) and average number of objective function evaluations (b).	111
4.6	Pairwise Spearman's rank correlation coefficients ρ_s between a range rewriting rules $N_P = \{2 \dots 6\}$ on Test Case 1. The ranks are ordered in Sub-figures a and b using the same measures as in Tables 4.5(a) and 4.5(b), respectively.	112
5.1	The ranges of the alternative baseline and linear age variables.	120

¹www.shapeways.com (accessed on 10th March 2016)

5.2	Results of the statistical study on the baseline and linear age parameterizations. The linear age parameterization is used with and without the local search. Results for completion rate p_c and average number of objective function evaluations, $\bar{Q}$, are presented.	121
5.3	Design space parameters of the sUAV wing topology optimization problem.	126
5.4	Comparison of the optimized internal structure mass, m_{IS} , and required objective function evaluations, $\bar{Q}$	127
6.1	Minimum and maximum values of additional variables and variables associated with parametric symbols.	141
6.2	Parameters defining the optimization problem.	147
6.3	The family of statistical tests ($j = 1 \dots 6$), ranked in decreasing order of significance. In each test, control parameters 1 has on average of the obtained results a better performance than control parameters 2. Based on the results, we reject the null hypothesis H_0 in tests $j = 1 \dots 4$	150
6.4	Optimization problem objectives and parameters k_p/k_0 and ϕ : in optimization problems #1-6 the objective is to minimize the average temperature $\bar{T}$, whereas in optimization problems #7-12 the objective is to minimize the maximum temperature $T_{\max}$	156
6.5	Results in numerical format, including the optimized average temperature $\bar{T}$ and the corresponding nondimensional thermal resistance $R_{\{\bar{T}\}}$, as well as the average number of required objective function evaluations, $\bar{Q}$. The results obtained by the L-systems-based method are reported as the mean and its 95% confidence interval. The estimate of required objective function evaluations of the direct encoding method is based on Section 2.4 and Figure 2 in Ref. Boichot and Fan (2016).	163
6.6	Results in numerical format (see the caption of Table 6.5 for explanations of the symbols).	164
7.1	Properties of the optimization problem.	172
7.2	Benchmarking of topology optimization results against the baseline design, i.e. the orthogrid with four stiffeners in x -direction. ω_f and Q are the optimized, fundamental natural frequency and the number of required objective function evaluations, respectively.	186
A.1	The family of statistical tests ($j = 1, 2, \dots, 30$) ranked in decreasing order of significance, based on their p -values. In each test, algorithm 1 has, on average of the obtained results, a better performance than algorithm 2.	196
C.1	Contributions to the work reported in Chapter 6.	199

Nomenclature

Abbreviations

AirCONICS	Aircraft Configuration through Integrated Cross-disciplinary Scripting
BESO	Bi-directional Evolutionary Structural Optimization
CFD	Computational Fluid Dynamics
CPU	Central Processing Unit
CPPN	Compositional Pattern Producing Network
CST	Class-Shape Transformation
DNA	Deoxyribonucleic Acid
FDR	False Discovery Rate
FVM	Finite Volume Method
FE	Finite Element
ES	Evolution Strategies
ESO	Evolutionary Structural Optimization
GA	Genetic Algorithm
GS	Ground Structure
L-Systems	Lindenmayer-Systems
LS	Linear Search
LSM	Level-Set Method
MDAO	Multidisciplinary Design, Analysis and Optimization
MDO	Multidisciplinary Optimization
MMA	Method of Moving Asymptotes
NEAT	NeuroEvolution for Augmenting Topologies
NESO	Nodal Evolutionary Structural Optimization
NSGA-II	Non-dominated Sorting Genetic Algorithm II
NURBS	Non-Uniform Rational B-Splines
OML	Outer Mold Line
RAM	Rapid Aircraft Modeler
SIMP	Solid Isotropic Material with Penalization
SPEA2	Strength Pareto Evolutionary Algorithm 2
sUAV	small Unmanned Aerial Vehicle
TC	Test Case

UAV	Unmanned Aerial Vehicle
VEGA	Vector Evaluated Genetic Algorithm
VLSI	Very-Large-Scale Integration
VSP	Vehicle Sketch Pad

Superscripts

dec	decrease
inc	increase
limit	maximum allowed value
max	maximum
min	minimum
rec	recovery of an element or a structural member
rej	rejection of an element or a structural member
T	transpose

Symbols

*	increase in the component hierarchy (map L-systems) [-]
+	right hand side marker (map L-systems) or anti-clockwise turn (turtle interpretation) [-]
−	left hand side marker (map L-systems) or clockwise turn (turtle interpretation) [-]
/	decrease in the component hierarchy (map L-systems) [-]
α	sensitivity number [-]
α_e	minimum edge angle in an offspring cell (map L-systems) [°]
α_s	significance level [-]
β	token (L-systems) [-]
Δ	difference
Γ	material boundary [-]
Γ_D	Dirichlet boundary
Γ_N	Neumann boundary
Γ_{wing}	dihedral [°]
Θ	ambient domain
θ	angle [°]
θ_0	initial heading (turtle interpretation) [rad]
Λ_{LE}	sweep angle at the leading edge [°]
λ	number of mutants in ES [-]
λ_1	lowest eigenvalue, critical buckling load [N]
$\bar{\lambda}_1$	normalized critical buckling load [-]
λ_j	the j th eigenvalue of a system, buckling load [F]
$\bar{\lambda}_{\text{ref}}$	normalized reference buckling load [-]
λ_t	taper ratio [-]
μ	mean

ν	Poisson's ratio [-]
ρ	density [kg/m ³]
$\bar{\rho}$	normalized density [-]
ρ_0	reference density [kg/m ³]
ρ_s	Spearman's rank correlation coefficient [-]
Σ	alphabet (L-systems) [-]
σ	standard deviation
σ_i	the i th letter of the alphabet Σ
σ_{vM}	von Mises stress [Pa]
σ_y	yield stress [Pa]
$\Phi(\mathbf{x})$	penalty function
ϕ	volume fraction [-]
ϕ_{ls}	level-set function
χ_{Ω_p}	characteristic function of domain Ω_p
Ψ	buffer domain
Ω	design domain
Ω_0	low conductive material domain
Ω_p	high conductive material domain
ω_0	axiom (the 0th developmental stage) [-]
ω_f	fundamental natural frequency [Hz]
ω_n	the n th developmental stage [-]
A	area [m ²]
A_{cs}	cross-sectional area [m ²]
A_k	area of the largest pentagonal shaped cell k in a phenotype [-]
$\mathcal{R}$	aspect ratio [-]
B_{nodes}	number of boundary nodes in a phenotype [-]
b	wing span [m]
c_b	scalar constant in the LSM [-]
c_{extent}	extent variable [-]
c_{lh}	lightening hole dimension fraction of the main dimensions of the member [-]
c_{FoS}	factor of safety [-]
c_M	proportion of individuals to be mutated [-]
c_m	element-specific mutation rate [-]
c_s	see symbol $@(c_s)$
c_{sca}	element width scaling factor [m]
$c_{t,j}$	trapezoidal variable, corresponding to the letter σ_j (turtle interpretation) [-]
c_w	see symbol $\&(c_w)$
c_x	crossover rate [-]
D_0	subdomains of domain Ω_p that satisfy the volume constraint
d	distance or dimension [m]
E	Young's modulus [Pa]

E_0	reference Young's modulus [Pa]
E_{bool}	elitism (boolean) [-]
E_u	union of octagonal elements (turtle interpretation)
g	gravitational acceleration [m/s ²]
$g(x, c^{\min}, c^{\max})$	scaling function [-]
F_{des}	design load [N]
f_a	minimum area fraction between parent and offspring cells (map L-systems) [-]
f_e	minimum edge fraction in an offspring cell (map L-systems) [-]
f_g	gravity load [N/m]
f_i	objective function of the i th test case [-]
$\bar{f}_{\min}$	average of optimized objective function values [-]
f_p	pressure load [N/m]
$f(\mathbf{x})$	objective function
H_0	null hypothesis
H_1	alternative hypothesis
$h_{\text{stiffener}}$	stiffener height [m]
i	index [-]
j	index [-]
$k(\mathbf{x})$	constraint function
$\mathbf{K}$	global stiffness matrix [N/m]
$\mathbf{K}_g$	global geometric stiffness matrix [1/m]
k	index [-]
k	conductivity [W/(m K)]
k_0	conductivity in domain Ω_0 [W/(m K)]
k_b	buckling penalty [-]
$\mathbf{k}_i$	stiffness matrix of an element i [N/m]
k_p	conductivity in domain Ω_p [W/(m K)]
$k_{s,i}$	stress penalty from the i th section [-]
L/D	lift-to-drag ratio [-]
l	length [m]
l_{branch}	maximum dimension of a turtle interpretation path (from its starting point) (turtle interpretation) [m]
$l_{\text{stiffener}}$	total length of stiffeners [m]
m	maturity [-]
m_e	mass of an material element [kg]
m_{wing}	wing mass [kg]
m_{IS}	internal structure mass [kg]
m_{panel}	total mass of an integrally stiffened panel [kg]
$m_{\text{stiffener}}$	total mass of stiffeners [kg]
m_{TO}	take of mass [kg]

$m \times n$	ground structure density [-]
M	material domain [-]
N	discrete quantity [-]
N_a	number of letters in an axiom (L-systems) [-]
N_{cells}	number of cells in a phenotype [-]
N_{elite}	number of elite individuals [-]
N_{gs}	number of ground structure members [-]
N_{nodes}	number of nodes in a phenotype [-]
N_P	number of rewriting rules (L-systems) [-]
N_{pool}	pool size in tournament selection [-]
N_{pop}	population size [-]
N_r	number of tokens in a rule (L-systems) [-]
N_w	number of non-dimensional widths [-]
n	age (L-systems) or discrete quantity [-]
n_c	number of columns in a matrix [-]
n_{con}	number of generations to convergence [-]
n_d	number of design variables [-]
n_k	number of constraints [-]
n_{lin}	linear age (L-systems) [-]
n_L	load factor [-]
n_{obj}	number of objectives [-]
n_r	number of rows in a matrix [-]
n_{runs}	number of optimization runs [-]
n_{tot}	the total number of possible recoveries from a full ground structure
n_x	number of grid elements in x -direction [-]
n_y	number of grid elements in y -direction [-]
P_A	penalty term [-]
P_N	penalty term [-]
P_i	i th rewriting rule (L-systems) [-]
p	probability [-]
$\hat{p}$	pooled completion rate [-]
p_c	completion rate [-]
p_{empty}	probability of a token to be empty (L-systems) [-]
p_{global}	probability of finding the global optimum [-]
p_{marker}	probability of a token to be a marker (L-systems) [-]
p_s	probability that a single step in a random walk is straight ahead [-]
Q	number of objective function evaluations [-]
$\bar{Q}$	average number of objective function evaluations [-]
Q_g	expected number of objective function evaluations to find the global optimum [-]
q	heat generation rate [W/m ²]

q_0	heat generation rate in domain Ω_0 [W/m ²]
q_p	heat generation rate in domain Ω_p [W/m ²]
$R_{\{T_{\text{ref}}\}}$	non-dimensional thermal resistance (with respect to the reference temperature T_{ref}) [-]
RR_j	rejection rate at iteration j
r	penalty coefficient [-]
r_b	buckling penalty coefficient [-]
r_s	stress penalty coefficient [-]
S	wing area [m ²]
S_e	standard deviation of edge lengths in a phenotype [-]
S_a	standard deviation of cell areas in a phenotype [-]
SE	standard error
$s_{\alpha,i}$	standard deviation of edge angles in the cell i [-]
$s_{e,i}$	standard deviation of edge lengths in the cell i [-]
$\bar{T}$	average temperature [K]
T_{max}	maximum temperature [K]
$T_{\text{max,DE}}$	optimized maximum temperature obtained by the direct encoding method [K]
T_{ref}	reference temperature [K]
$\bar{T}_{\text{SIMP}}$	optimized average temperature obtained by the SIMP method [K]
T_{sink}	sink temperature [K]
t_{skin}	skin thickness [m]
$t_{\text{stiffener}}$	stiffener thickness [m]
$\mathbf{u}$	displacement vector [m]
V	volume [m ³]
w	width [m]
w_{frac}	fraction between non-dimensional widths [-]
X_{type}	crossover type
x	material element (SIMP) or point in a design domain
x_i	i th element of the genotype
$\mathbf{x}$	vector of design variables, i.e. a genotype
y_0	vertical coordinate of the initial orientation (turtle interpretation) [m]
[	start of a marker (map L-systems) or a branch (turtle interpretation) [-]
]	end of a marker (map L-systems) or a branch (turtle interpretation) [-]
$\$(\theta)$	turn by angle θ (parametric L-systems) [°]
@(c_s)	multiply the prevailing step size by c_s (parametric L-systems) [-]
&(c_w)	multiply the prevailing step size by c_w (parametric L-systems) [-]

Chapter 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 contains 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 organism¹. The compact format of storing genetic information has enabled nature to gradually evolve intriguingly complex organisms.

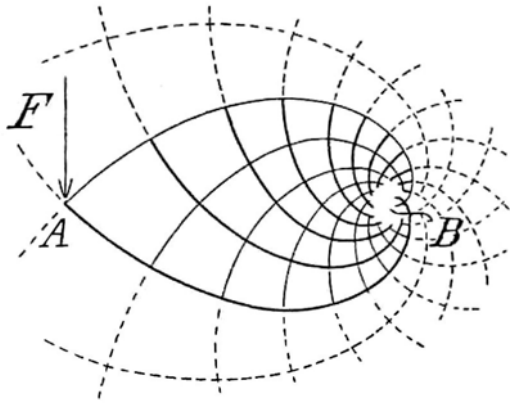
In evolutionary algorithms, the genotype is the numerical representation of a solution candidate, whereas the phenotype is its corresponding ‘physical’ instance. However, in the majority of studies applying evolutionary algorithms, the genotype consists of design variables that explicitly define units of the phenotype, referred to as *direct encoding*. This approach ignores the developmental aspect of the biological genotype-phenotype distinction. Alternatively, *generative encodings*² are parameterization methods that implicitly define units of the phenotype via a developmental recipe (in the same way as the DNA sequence does). 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, Kobayashi et al., 2010). Although generative encodings have been demonstrated to outperform direct encodings on some applications (Hornby and Pollack, 2001, 2002, Pedro and Kobayashi, 2011), their full potential, or limitations, have not yet been thoroughly studied.

In fact, designs consisting of repeating patterns of similar structural members are often desired in engineering. They can be seen for example in the well-known minimum mass

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

²Also referred to as *developmental encodings* or *artificial embryogeny*.

truss layout derived by [Michell \(1904\)](#) (Figure 1.1(a)), the Sydney Opera House (Figure 1.1(b)) and the Astoria-Megler Bridge, spanning over the Columbia River (Figure 1.1(c)). Each of these examples have a different motivation for similar structural members. [Michell \(1904\)](#) proved that the minimum mass of a truss structure, made from material with equal or unequal allowed stresses in tension and compression, is obtained using two similar logarithmic spirals³. The architects of the Sydney Opera House designed its roof to consist of similar shell structures for aesthetic reasons. Finally, a key aspect in the design of Astoria-Megler Bridge is its manufacturability, in which similar structural members are beneficial. Generative encodings have a natural tendency to yield designs with these kind of repeating patterns of similar structural members.



(a) Michell truss ([Michell, 1904](#))



(b) Sydney opera house (Photo: Pixabay / Anna Mustermann)



(c) Astoria-Megler truss bridge (Photo: Pixabay / Dovid Smith and Niccolea Miouo Nance)

Figure 1.1: Designs consisting of similar structural members.

One of the research fields in structural engineering is topology optimization, which comprises search methods of seeking the optimal material distribution in a given design domain. Commonly used gradient-based methods, such as Solid Isotropic Material with Penalization (SIMP) ([Bendsøe, 1989](#)) and Evolutionary Structural Optimization (ESO)

³Logarithmic spirals (also equiangular spirals) are curves, often appearing in nature, the sections of which are self-similar.

(Xie and Steven, 1992), 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 the design domain, and the optimal subset of these members is sought.

Several evolutionary algorithms (which do not use gradient information) have also been applied to topology optimization, using either direct or generative encodings. Evolutionary algorithms have the following advantages. First, they are considered as global optimization methods, which are able to operate in design landscapes with multiple local optima (Keane and Nair, 2005). Second, they are well-suited to ‘black box’ type optimization problems (Weise, 2009), where no gradient information is available for the objective function. These optimization problems include objective functions that are stochastic, integer-valued and implicitly defined. The aforementioned generative encodings result in implicitly defined objective functions. Third, evolutionary algorithms are particularly suitable for multi-objective optimization because they simultaneously evolve a population of candidate designs (Coello Coello et al., 2007). Thus, a single optimization run with a multi-objective evolutionary algorithm yields a set of designs that represent an approximation of the frontier of the best trade-off solutions to the objectives, referred to as the *Pareto front*. As a comparison, discovering the approximated Pareto front using gradient-based methods typically requires a series of optimization runs with aggregated objective functions.

Despite these advantages, evolutionary algorithms have only gained limited acceptance in the topology optimization community. Munk et al. (2015) list two partial reasons for this, which are the difficulty of ensuring structural connectivity and the excessive use of computational resources. Sigmund (2011) indicates that non-gradient-based optimization methods (which include evolutionary algorithms) require orders of magnitude more function evaluations in comparison to gradient-based methods.

1.1 Aims of the research

This work has two primary aims, which are defined in the following. The first is to find improvements in the *performance* of evolutionary topology optimization algorithms, which use either direct or generative encodings. We evaluate the performance of an algorithm based on its effectiveness (its ability to find good solutions) and efficiency (its ability to find them quickly). Therefore, the aim is not only dedicated to mitigate the excessive use of computational resources, as indicated by Munk et al. (2015) and Sigmund (2011), but also to improve the goodness of the solutions.

The second aim is to identify topology optimization problems in engineering design, to which generative encodings are particularly suitable – or the contrary, particularly unsuitable. Thus, the objective is to select a diverse set of engineering design problems, to which evolutionary topology optimization algorithms with both direct and generative encodings are applied. In addition, the purpose is to select such engineering design problems that the algorithms we develop in this work have the potential to be useful for practitioners working in the corresponding fields. Later, in Section 2.9, we will define six hypotheses that correspond to these two primary aims.

Both evolutionary algorithms and generative encodings are research fields with numerous classes. We narrow the scope of our research as follows, in order to keep its size manageable. First, we narrow evolutionary algorithms to genetic algorithms (GAs), as they are described as the most prominent and widely used class of evolutionary algorithms (Sivanandam and Deepa, 2007). Second, we narrow generative encodings to those based on L-systems (Lindenmayer, 1968a,b), which have recently gained popularity among the topology optimization researchers.

1.2 Structure of the thesis

The content of the thesis is the following. Chapter 2 reviews the literature of topology optimization, with an emphasis on evolutionary algorithms and their encoding methods. In addition, three areas of application are selected and reviewed. At the end of the chapter, research hypotheses and objectives are defined in order to achieve the first aim of the research: finding improvements to the performance of evolutionary topology optimization.

The purpose of Chapters 3 to 5 is to conduct the work of these research objectives on direct and generative encodings. The proposed improvements are tested on the first application, i.e. mass minimization of the wing internal structure of a small Unmanned Aerial Vehicle (sUAV).

In Chapters 6 and 7, we continue to apply the methods to the second and third application. These applications are temperature minimization of a conductive heat transfer system and natural frequency maximization of an integrally stiffened panel structure, respectively. In these chapters, we apply methods using both direct and generative encodings to evaluate the second aim of the research: identifying topology optimization problems in engineering design, to which generative encodings are particularly suitable.

Finally, Chapter 8 concludes the work and provides recommendations for the future work.

Chapter 2

Topology optimization

Topology optimization is the process of determining the optimal distribution of material inside a predefined design domain, in order to minimize, or maximize, a physical quantity of the resulting design. The search space in topology optimization has greater design freedom in comparison to other types of structural optimization, e.g. shape and size optimization. The reason is that topology optimization may flexibly vary the general configuration of the structure, whereas shape and size optimization are restricted to a single predefined configuration. Currently, topology optimization is the most active research area of structural and multidisciplinary optimization ([Deaton and Grandhi, 2014](#)).

In this chapter, we review first commonly used design space parameterizations (Section [2.2](#)) and search algorithms (Sections [2.3-2.6](#)) in topology optimization. Then, we review L-systems, the type of generative encoding we use in this work, in detail in Section [2.7](#). Finally, three areas of application are selected and reviewed in Section [2.8](#). However, before all this, let us start by a short introduction to optimization.

2.1 Introduction to optimization

Optimization features extensively in nature and engineering. Migrating geese fly in a flock to maximize their range. Evolution drives a population towards a better adaptation to the environment. Engineers design and develop their products to have as high performance, reliability or cost-efficiency as possible. Optimization among human and other species is either intentional or unintentional. Science and engineering include a wide range of both optimization problems and methods to seek the optimal solution.

From a mathematical point of view, optimization means either the minimization or maximization of an objective function subject to given constraints, and with respect to predefined design variables ([Nocedal and Wright, 2006](#)). Before selecting a suitable

optimization method, the optimization problem must be classified according to multiple criteria. In the following, four, perhaps the most important, criteria are presented. First, design variables are either continuous or discrete. Second, the optimization problem can be constrained or unconstrained. Third, a local or the global optimum may be desired, depending on the purpose. Fourth, the optimization problem might have single or multiple objectives.

Optimization methods can be categorized based on their approach to the optimization problem; these categories are gradient-based, rule-based and stochastic methods (Keane and Nair, 2005). Gradient-based methods sample points in the design space and use the gradient information of the current point, or the current and past points, to search for better optimized points. A variety of gradient-based methods have been developed, such as the steepest-descent direction and conjugate gradient methods (Nocedal and Wright, 2006). A common feature for all of them is that they require the gradient information in the whole search space. Gradient-based methods are very effective in local optimization with continuous design variables. However, they are not suitable for optimization with discrete design variables, and they might only find a local optimum for a global optimization problem.

Rule-based methods are structured sampling strategies that aim to bound the optimum inside a limited space, and then reduce the size of the space until the optimum has been narrowed down to a specified precision. The simplest examples of these methods are the golden section and the Fibonacci search. Unlike gradient-based methods, rule-based methods are well-suited to optimization with discrete variables. However, they may still fail to find the global optimum.

Stochastic methods sample the objective function by generating and using random variables. Optimization procedures started from the same exact point follow a different optimization trajectory if the sequence of random numbers is varied. Considering a minimization problem, if the objective function value of a certain sampled point is low in comparison to other sampled points, the region of the point is likely to be further sampled during the next iterations. Typical examples of stochastic optimization methods are evolutionary and swarm algorithms. They are suitable for continuous and discrete optimization variables, but may require a large number of function evaluations.

For a more detailed review of optimization methods, the reader may wish to consult optimization textbooks, such as by Nocedal and Wright (2006), Fletcher (2013) and Simon (2013), or the textbook by Keane and Nair (2005), which reviews optimization methods from an aircraft design point of view.

2.2 Design space parameterization

The first step of any topology optimization process is establishing the design space. In standard continuous optimization problems, this is simply a matter of defining the ranges of the design variables, but the space of topologies has no conventional design variables and thus no ranges either, as defined in the conventional sense. Thus, the first step is to define the mapping between the design variables and the material distribution in the design domain, referred to as the *parameterization* (also *design representation*).

We here categorize parameterization methods into direct, geometric and generative encodings. The categorization is basically the same as the one defined by [Aulig and Olhofer \(2016\)](#). However, the authors use the terms grid, geometric and indirect representations, but we choose to follow here the taxonomy often used in the context of generative encodings, i.e. direct and generative encodings, to which we add geometric encodings.

2.2.1 Direct encodings

Ideally, the best design would be obtained by picking the best combination of structural members from the *structural universe*, which is an infinite collection of all permissible structural members that a design may contain. In reality, this is not possible because an infinite number of objective function evaluations would be needed. However, the structural members may be picked from a *ground structure*, first introduced by [Dorn et al. \(1964\)](#), which is a finite, but large, subset of the structural universe. Each of these ground structure members is explicitly assigned a binary design variable, and the optimization problem is defined as finding the optimal subset of the ground structure members. We will return to the ground structure approach in Section 2.4.

Another approach to establishing the design space parameterization is to distribute homogeneous material inside throughout the design space and discretize the material into finite elements, a method first presented by [Bendsøe and Kikuchi \(1988\)](#) and referred to as the *continuum material* or *grid representation*. Each element is then assigned a design variable(s) describing its presence in the design. The design variable may be a binary value, describing strict 0/1 material distribution, or a scalar value, allowing the density of the element to vary between 0 and 1. An optimization method for the binary variables is presented in Section 2.5 (Evolutionary Structural Optimization (ESO)), and for scalar variables in Section 2.6.1 (Solid Isotropic Material with Penalization (SIMP)).

2.2.2 Geometric encodings

In geometric encodings, the phenotype is defined via movable shape primitives, such as position, shape and thickness ([Aulig and Olhofer, 2016](#)). Depending on the requirements

of evaluating the objective function, the geometric phenotype may be further mapped into a grid representation. Unlike in direct encodings, the number of design variables in geometric encodings is independent of the design resolution.

Various types of geometric encodings have been suggested in the literature, of which we list a couple of examples in the following. [Sethian and Wiegmann \(2000\)](#) represented their designs via a scalar level-set function ([Osher and Sethian, 1988](#)). We will review the method, referred to as the *Level-Set Method* (LSM), in more detail in [Section 2.6.2](#). [Schoenauer \(1996\)](#), [Hamda and Schoenauer \(2002\)](#) represented their phenotypes via Voronoi diagrams¹, where each of the sites is assigned a binary design variable defining the existence of material inside its region. [Tai and Chee \(2000\)](#) parameterized the design space via a set of Bézier curves. The design variables were the location of the control points defining the Bézier curves and additional thickness variables. For a more detailed review of geometric encodings, the reader may wish to consult the paper by [Aulig and Olhofer \(2016\)](#).

2.2.3 Generative encodings

In generative encodings, a genotype comprises rules or biological processes that implicitly define the material distribution in the corresponding phenotype. As indicated in the introduction, this approach enables repeated usage of genotype elements, and therefore more compact representation of phenotypes than direct and geometric encodings ([Hornby and Pollack, 2001](#), [Stanley and Miikkulainen, 2003](#), [Kobayashi et al., 2010](#)).

Generative encodings typically mimic the morphogenesis of biological organisms. However, as the developmental processes in the nature are extremely complex, scientists have chosen to use various levels of abstraction, i.e. how precisely the natural processes are mimicked. [Stanley and Miikkulainen \(2003\)](#) categorize two levels of abstraction into *grammatical* and *cell chemistry approaches*.

Grammatical approaches are high-level abstractions that trace back to the mathematical models by [Lindenmayer \(1968a,b\)](#) (L-systems). In fact, these models were not originally intended to be used in topology optimization, but to resemble the developmental processes of living organisms, e.g. bacteria and plants. The idea is that organisms, which may have complex geometries, are modeled by repeatedly modifying a simple starting object by a set of predefined rules. When used in topology optimization, the idea is to evolve the set of rules and the starting object to minimize or maximize a quantity related to the corresponding phenotype. Examples of grammatical approaches are cellular encoding ([Gruau, 1993, 1994](#)) and edge encoding ([Luke and Spector, 1996](#)). However, the generative encodings we use in this work are directly based on L-systems.

¹A Voronoi diagram is a partitioning of a plane into regions using control points, which are referred to as *sites*. Each site has a corresponding region, which includes all points in the plane that lay closer to the site than any other site.

We will review both L-systems and their applications to topology optimization in detail in Section 2.7.

Cell chemistry approaches are low-level abstractions that closely mimic the chemical interactions inside and between cells during embryogenesis (Stanley and Miikkulainen, 2003, Stanley, 2007). These approaches evolve genes that produce proteins, which guide the cell division process as the phenotype grows. Figure 2.1 shows an example growth procedure of a design, modeled using such an approach.

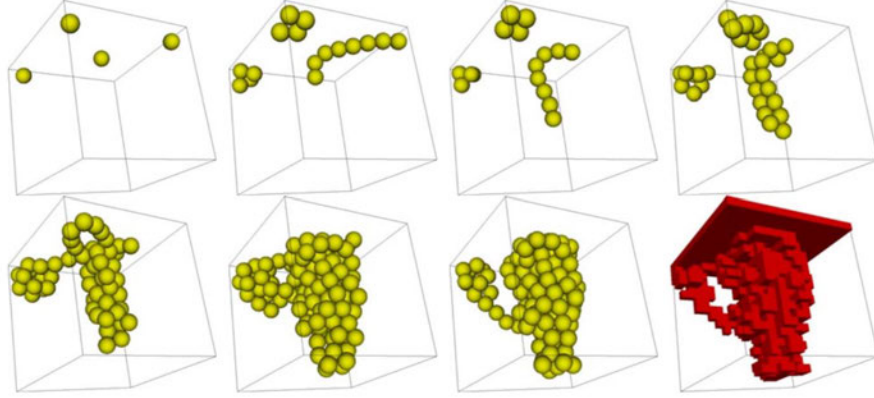


Figure 2.1: Construction of a phenotype in stages using a cell chemistry approach, based on motile polarized cells (Steiner et al., 2009). The last figure represents discretization into a three-dimensional grid.

Another approach outside these two categories is worth mentioning: the *Compositional Pattern Producing Network* (CPPN). In all of the above mentioned generative encoding approaches, phenotypes are constructed via temporal, intermediate stages of the development (see Section 2.1). These stages are useful in understanding the bio-inspired aspect of generative encodings. However, in generative encodings, the abstraction is not required to explicitly follow natural processes. Stanley (2007) proposed the CPPN, which is free of temporal stages. The approach is a variation of artificial neural networks, in which each of the nodes in the network may contain a variety of mathematical functions, e.g. gaussian or trigonometric functions (Figure 2.2). The distribution of material in the phenotype is then defined via this network of functions. The topology of the network and the content of its nodes are typically evolved by NeuroEvolution for Augmenting Topologies (NEAT) (Stanley and Miikkulainen, 2002).

Later in this work, we will use the ground structure approach and the L-systems-based methods, which both may be evolved via genetic algorithms (GAs). The next section presents GAs in terms of their encoding, genetic operations, as well as reviews constrained and multi-objective GAs.

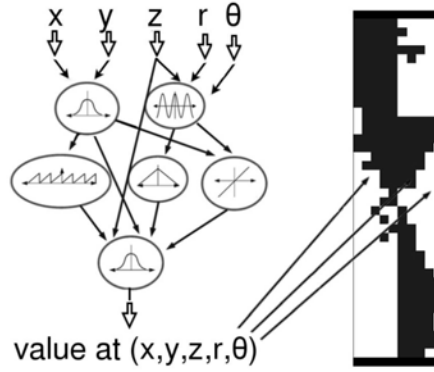


Figure 2.2: An illustration of the CPPN method to encode a phenotype, adopted from the paper by [Cheney et al. \(2014\)](#).

2.3 Genetic algorithms (GAs)

GAs are optimization methods that mimic Darwinian natural evolution. In GAs, a solution candidate is encoded as a sequence of integer or real numbers, i.e. the genotype. GAs iteratively evolve a population of these genotypes towards the best adoption to the simulated environment (which merit is the objective function), using mathematical operators mimicking natural selection, recombination and mutation. As optimization methods, GAs are described as gradient-free global search methods that perform well on non-differentiable functions and functions with many local optima ([Whitley, 1994](#)).

GAs are a subset of evolutionary algorithms, which initiated in the 1950s and 1960s when scientists studied how the evolution theory could be exploited as an optimization method. [Rechenberg \(1965, 1973\)](#) presented the first evolution strategies (ESs), which were further improved by [Schwefel \(1975, 1977\)](#). In a simple form of the algorithm, a parent produces λ mutants, inheriting their elements from the parent with some random variations (i.e. mutations). The fittest individual among offspring, or among parents and offspring, is selected to become the new parent. The former is referred to as $(1, \lambda)$ -ES, and the latter as $(1 + \lambda)$ -ES. A genotype in ESs typically is a vector of real numbers. GAs, in which we focus on in the following, were developed by [Holland \(1975\)](#) and his students, such as [De Jong \(1975\)](#).

[Koza \(1992, 1994\)](#) developed *genetic programming* as an extension to Holland's work on GAs. Instead of encoding the genetic information on a fixed-length sequence of integer/scalar numbers, genetic programming uses tree structures, encoded into variable length genotypes, to represent solution candidates. Similar to GAs, these trees are ranked based on their fitness and genetic operations are applied to the selected trees. All above mentioned branches, as well as several other branches, of evolutionary algorithms are still active fields of research.

2.3.1 Encoding

As mentioned earlier, genotypes in a GA carry the genetic information of the population. While the DNA sequence in nature is based on a four-character encoding, the genetic information in a GA is typically encoded using binary values (Mitchell, 1998, Sivanandam and Deepa, 2007) though also multiple character and real number encodings exist. The reason for the popularity of binary strings is perhaps historical – Holland and his students used binary values in their early work (Mitchell, 1998). Holland (1975) compared two types of encodings that can store the same amount of genetic information: a long string encoded using a small number of characters and a short string using a large number of characters. He argued that the former is able to evaluate a higher number of possible bit combinations, i.e. schemata², in comparison to the latter. Thus, the performance of the algorithm would be better with a small number of characters. However, opposite indications have been obtained in some later studies. For example, Janikow and Michalewicz (1991) observed the real value encoding to be faster and more consistent than the binary encoding on their test case. Simon (2013) presents an example where the real number encoding outperforms the binary encoding on the two-dimensional Ackley function. The optimal encoding method is likely to be dependent on the optimization problem.

Design variables in a GA, which may be binary, integer or scalar values, are encoded into genotypes. The encoding of binary and integer variables is relatively simple. However, an issue might appear on integer variables. Let us consider an integer variable with 300 possible values. The closest binary lengths are eight ($2^8 = 256$) and nine bits ($2^9 = 512$). The eight-bit string has obviously too few ‘slots’ to accommodate all 300 values, whereas the nine-bit string would have 212 excess ‘slots’ that are meaningless and might mislead the optimization process. Scalar design variables are encoded either as integer or real numbers. In the case of the former, a scalar variable must be discretized inside relevant bounds. Each discretized point is represented by a sequence of integer values. The smoothness of the discretization is to be chosen so that the optimum can be located with adequate accuracy. However, the smoother the discretization is, the more elements are required in the genotype to represent the discretized points of scalar design variables, and thus, the more function evaluations are required to obtain a converged solution.

2.3.1.1 Multi-dimensional genotypes in direct encoding

Another aspect to be considered is the choice between one- and multi-dimensional encodings. In a classical one-dimensional encoding, the genotype is a vector containing the elements with genetic information, whereas in a two-dimensional encoding the elements are arranged in a matrix form. The vast majority of GA-based optimization applications

²Schemata are patterns of fixed and variable bit combinations that are favored by evolutionary processes.

use one-dimensional encoding, while the use of two-dimensional (or higher) encodings is very rare. However, the physical design domains of many practical problems are two- or three-dimensional (Gen and Cheng, 2000). Bui and Moon (1995) argue that if such a problem is encoded into a one-dimensional genotype, a considerable amount of geometric information is lost. The benefit of multi-dimensional encoding is in the preservation of the *geographical linkage* between the elements. The geographical linkage means that two elements located close to each other in a genotype are more likely to survive together to the next generation than two randomly selected elements in the same genotype.

The first application of two-dimensional encoding is due to Cohoon and Paris (1987), who applied it to optimization of a Very-Large-Scale Integration (VLSI) circuit. Later, two-dimensional encoding has been used for example in job scheduling (Chou et al., 2008, Ono et al., 1996), and graph partitioning problems (Kim et al., 2011). Giger and Ermanni (2006) used the so-called ‘graph-based parameterization’ in evolutionary optimization of the topology, shape and size of a truss structure. The authors encoded the element connectivity of the nodes on two-dimensional genotypes. Considering the two-dimensional nature of the internal structure arrangement inside a wing, GA-based optimization could perhaps be improved by using two-dimensional encoding. However, two-dimensional encoding has not yet been applied to the topology optimization of an aircraft wing (as far as were able to ascertain).

2.3.2 Genetic operators

In the Darwinian natural evolution, the fittest individuals survive to reproduction, where the genes of, usually, two individuals are recombined. During the process, some of the genes are randomly mutated. In GAs, these processes are referred to as operators, namely the *selector*, *crossover* and *mutator*. Multiple algorithms have been developed for each operator. In the following, we describe commonly-used algorithms.

In the *roulette wheel* selector, the individuals are assigned a sector of a roulette wheel, proportional to their fitness. The wheel is spun, and the resulting sector determines the individual that is selected for reproduction. The process is repeated n times, where n is the number of individuals in the population. The roulette wheel selector has a risk of premature convergence in a situation where an early population has a few superior individuals in comparison to the others (Mitchell, 1998). With significantly better chance to reproduce, the fittest individuals, and their offspring, may quickly fill the population and cause premature convergence.

The *rank selector* could be used to avoid premature convergence. In the rank selector, individuals in the population are first ranked based on their fitness, and then assigned a survival probability based on their rank. The rank selector preserves diversity in the population, but may retard the optimization process, due to the reduced selection

pressure. Another commonly used selector is the *tournament selector*, in which a number of individuals, N_{pool} , are randomly selected from the population into a tournament pool, and the best individual in the pool is selected to reproduction. The selection pressure of the tournament selector can be varied by changing the size of the pool (the pool size $N_{\text{pool}} = 2$ induces the lowest selection pressure). A benefit of the tournament selector is in its computational efficiency because no ranking of the population, nor preparation of the roulette wheel, is required. The tournament selector can also be used together with the roulette wheel to increase the selection pressure of the latter. In this version, the tournament pool is first filled using the roulette wheel selector, and then the best individual in the pool is selected for reproduction.

Occasionally, the best individual of a generation may become less fit than the best individual of the previous generation. In this situation, the best individual has not been selected for the reproduction or it is destroyed by a crossover or a mutator operator. A concept called *elitism*, first proposed by De Jong (1975), prevents the evolutionary process from losing its best individual(s). Elitism may be implemented in several ways, from which we describe two commonly used approaches. In the first, a new generation of N_{pop} individuals is formed by merging N_{elite} best individuals of the previous generation with $(N_{\text{pop}} - N_{\text{elite}})$ newly produced individuals. In the second, N_{pop} new individuals are first produced and evaluated, and then the worst individuals of the new generation are replaced by the best N_{elite} individuals of the previous generation. Many researchers indicate an improved performance of GAs when applying elitism (Mitchell, 1998), and, further, out of the two described elitist approaches, better performance is usually expected using the latter (Simon, 2013). However, in the very problem-dependent world of GAs, the scope of such conclusions is generally limited.

After selecting the individuals for reproduction, a crossover is applied on pairs of selected individuals. In the following, the crossovers are presented separately for one- and multi-dimensional encodings (see Section 2.3.1). Since three-(and above)dimensional encodings are rare, we only review two-dimensional crossovers as multi-dimensional crossovers. Two individuals, selected to mate, are referred to as the *parents*, and the resulting two new individuals as the *offspring*.

For one-dimensional encodings, the simplest crossover is the *single-point crossover*, where the genotypes of the parents are spliced at a random location. The offspring are formed by merging the first part of the genotype of the first parent with the second part of the genotype of the second parent, and visa versa. A drawback with the single-point crossover is that element combinations located close to the two extremes of the genotype are nearly always destroyed. An alternative crossover is the *two-point crossover*³, where the splicing is performed at two random locations of the genotype. In the two-point crossover, the end points of the genotype have the same probability of remaining together as any two adjacent elements in the genotype. In addition, the two-point crossover has

³Also referred to as the *ring crossover*.

greater splicing diversity than the single-point crossover. The splicing diversity may be even further increased by using the *uniform crossover*⁴, where each element in the first offspring is inherited from the first parent with the probability $p = 0.5$ and from the second parent with the complement probability $1 - p$. The second offspring is an inverse of the first. However, the preservation of geographical-linkage is poor in the uniform crossover.

While the crossover methods for one-dimensional encodings have become established, the methods for less used two-dimensional encodings vary in the literature. The simplest two-dimensional crossover is a linear splicing, where a two-dimensional genotype is spliced either horizontally or vertically between rows or columns of the genotype, respectively. The weakness in the linear splicing of a two-dimensional genotype is the low diversity of resulting splicing strategies. Let us consider a one-dimensional genotype having a length of $N = n^2$ and a two-dimensional genotype having a size of $n \times n$. The single-point crossover applied on the one-dimensional genotype yields $n^2 - 1$ possible splicing strategies, but the linear splicing crossover applied on the two-dimensional genotype, having the same number of elements, yields only $2(n - 1)$ splicing strategies (Kahng and Moon, 1995). The diversity of splicing strategies may be increased by increasing the complexity of the crossover. Examples of more complex crossovers for two-dimensional genotypes are found in the studies by Anderson et al. (1991), Jung and Moon (2002) and Sadrzadeh (2012). The uniform crossover on two-dimensional genotypes is equivalent to the same crossover on one-dimensional genotypes.

Mutators in GAs are considered as secondary operators for selectors and crossovers (Golberg, 1989). Their purpose is to prevent the optimization from converging prematurely to a local minimum by introducing random changes in single elements of the genotype. The *flip mutator* in binary value encodings simply changes an element from 0 to 1, or vice versa, whereas in integer value encodings the flip mutation randomly replaces the character with any possible character. The *Gaussian mutator*, suitable for real value encodings, adds a random Gaussian-distributed⁵ value to the element to be mutated. Alternatively, the *swap mutator* changes the locations of two randomly selected elements with each other. Mutators in one- and two-dimensional encodings work in the same way.

2.3.3 GAs with constraints

Many real-life optimization problems have constraints. Depending on the type of the constraint(s), different methods exist to handle them in GAs⁶. A simple bound of a scalar design variable or a range of integer design variable can be implemented in the encoding. However, the constraints might also be highly nonlinear and/or depend on many design

⁴Also referred to as the *distributed crossover*.

⁵Typically, the mean of the Gaussian distribution $\sigma = 0$ to avoid the drift of the element values.

⁶In fact, the methods presented in this section are applicable to evolutionary algorithms in general, but we here refer to GAs because of the scope of the current work.

variables. The simplest way of handling these constraints in a GA is to penalize the fitness of constraint violating designs to the extent that they are guaranteed not to survive. However, if multiple constraints exist, finding a feasible design might be nearly as difficult as finding the optimum, which may well make this approach impractical. Discarding the infeasible offspring also reduces the diversity of the population, which might lead to premature convergence to a local optimum.

Constrained optimization includes both equality and inequality constraints. Equality constraints can usually be implemented in the encoding, but inequality constraints need a special treatment.

The most common approach is to translate the constrained optimization problem into an unconstrained one by applying a penalty function. [Courant \(1943\)](#) used penalty functions first to solve differential equations, and [Fiocco and McCormick \(1968\)](#) were the first to apply the method to nonlinear optimization problems ([Joines and Houck, 1994](#)). In order to illustrate the idea of a penalty function, let us consider a generic optimization problem, with n_k inequality constraints and n_d design variables, defined as

$$\begin{aligned} & \text{minimize} && f(\mathbf{x}) \\ & \text{w.r.t} && x_i \quad i = 1, \dots, n_d \\ & \text{subject to} && k_j(\mathbf{x}) \geq 0 \quad j = 1, \dots, n_k, \end{aligned} \quad (2.1)$$

where $f(\mathbf{x})$ and $k_j(\mathbf{x})$ are the objective and constraint functions, respectively. The corresponding unconstrained optimization problem can be defined as

$$\begin{aligned} & \text{minimize} && f(\mathbf{x}) + r \sum_{j=1}^{n_k} \Phi[k_j(\mathbf{x})] \\ & \text{w.r.t} && x_i \quad i = 1, \dots, n_d, \end{aligned} \quad (2.2)$$

where Φ is the penalty function and r is the penalty coefficient ([Golberg, 1989](#)). Penalty function methods may use death, static, dynamic, annealing, adaptive or co-evolutionary penalties, or segregated GA ([Yeniay, 2005](#)). The practical problem in implementing such penalty functions is the choice of penalty coefficient r , which should scale the penalty term to have a similar amplitude as the objective function term.

In addition to penalty functions, the constraints in GAs may also be handled using the following categories of methods. *Special representations and operators* aim to eliminate infeasible regions from the design space. *Repair algorithms* are used to change infeasible individuals into feasible via a greedy search or a user-defined heuristic (the latter requires detailed understanding of the optimization problem). In *multi-objective optimization techniques*, the single-objective optimization problem with constraints is translated into an unconstrained multi-objective optimization problem, with $n_k + 1$ objectives, where n_k is the number of constraints. The resulting optimization problem is then solved using a multi-objective optimization algorithm (we provide a short review of multi-objective

GAs in the next section). In *hybrid methods*, the constraints are handled by combining the algorithm with another technique, often outside evolutionary algorithms, such as Lagrangian multipliers or fuzzy logic.

Coello Coello (2002) indicates in his review paper that penalty function methods are a good starting point for constrained evolutionary optimization. For more specific optimization problem types, he gives the following recommendations of the suitable constraint handling method:

- repair algorithms for combinatorial optimization problems,
- special representations and operations for optimization problems with linear constraints, and
- multi-objective optimization techniques for optimization problems with highly constrained search spaces.

2.3.4 Multi-objective GAs

The design of engineering systems is often multi-objective. Examples of design objectives are maximizing the performance, maximizing the reliability or minimizing the cost of the system. These objectives are nearly always conflicting, which means that no single design exists that is optimal in terms of all objectives.

A general optimization problem with n_{obj} objectives is defined as⁷

$$\begin{aligned}
 & \text{minimize} && \{f_1(\mathbf{x}) \dots f_{n_{\text{obj}}}(\mathbf{x})\} \\
 & \text{w.r.t} && x_i && i = 1, \dots, n_d \\
 & \text{subject to} && k_j(\mathbf{x}) \geq 0 && j = 1, \dots, n_k.
 \end{aligned} \tag{2.3}$$

A solution to the optimization problem is a set of *non-dominated* designs. Design $\mathbf{x}_1$ dominates design $\mathbf{x}_2$ if $f_i(\mathbf{x}_1) \leq f_i(\mathbf{x}_2)$ for all $i \in [1 \dots n_{\text{obj}}]$ and $f_i(\mathbf{x}_1) < f_i(\mathbf{x}_2)$ for at least one of the n_{obj} objectives. A design is non-dominated if it is feasible, i.e. it satisfies all n_k constraints, and no other known feasible design dominates it. The optimized set of non-dominated designs represents an approximation of the Pareto front, which consists of all feasible designs in the objective space, which are not dominated by any other feasible design.

Finding all designs that belong to the Pareto front is practically impossible in most real-life optimization problems. Nevertheless, an approximation of the Pareto front can be obtained using multi-objective optimization heuristics, which are desired to yield results

⁷We here define all objectives to be of the minimization type. A maximization objective can be included in the definition simply by multiplying the objective function by -1.

with the following conflicting goals (Zitzler et al., 2000, Konak et al., 2006, Zavala et al., 2014):

1. The approximated Pareto front should be as close to the true Pareto front as possible.
2. The optimized set of non-dominated designs should be distributed diversely along the approximated Pareto front (typically a uniform distribution is desired).
3. The approximated Pareto front should be able to capture the entire Pareto front, including its extremes.

Multi-objective GAs are well-suited for the purpose, as they are able to simultaneously search different regions of the objective space, even if it is non-convex, discontinuous or multi-modal. As a result, the designer obtains a set of optimized non-dominated designs. Therefore, the designer is not required to weight the objectives prior the optimization (though the interpretation of the result will require the expression of a particular bias).

The first multi-objective GA is due to Schaffer (1985) and is referred to as the Vector Evaluated Genetic Algorithm (VEGA). Thereafter, several multi-objective evolutionary algorithms were developed. In comparison to standard single-objective algorithms (Section 2.3.2), these algorithms typically use specialized selectors, which favor individuals closest to the true Pareto front, and *density estimators*, which enable the algorithms to direct the search into sparsely populated regions of the objective space. These features enhance the probability of the algorithm to achieve the three above listed goals.

One of the most popular multi-objective GAs is the NSGA-II (elitist Non-dominated Sorting Genetic Algorithm) by Deb et al. (2002). Zavala et al. (2014) indicate in their review paper that the algorithm has become the *de facto* multi-objective optimization heuristic. A characteristic feature of NSGA-II is its fast sorting procedure to assign individuals into non-domination levels. The first non-domination level consists of non-dominated individuals in the entire population, the second of non-dominated individuals in a sub-population, from which the first non-domination level is removed, and so on. The non-dominance level is used as the primary selection measure. The secondary selection measure is the *crowding distance*, which is the density estimator of the NSGA-II. The crowding distance of design $\mathbf{x}_i$ is a non-dimensional distance of its two neighbor designs, which belong to the same non-dominance level as design $\mathbf{x}_i$, in the objective space. The secondary measure is used as a tie breaker, if primary measures of two (or more) individuals are equal.

Another popular multi-objective GA worth mentioning is the SPEA2 (Strength Pareto Evolutionary Algorithm 2) by Zitzler et al. (2001). For an extensive list of multi-objective GAs, the reader may wish to consult review papers by Konak et al. (2006) and Zavala et al. (2014).

2.4 Ground structure approach

In this, and the two following sections, we review popular topology optimization methods in the literature.

The ground structure approach, first introduced by [Dorn et al. \(1964\)](#), is a popular topology optimization approach, especially for truss structures. The ground structure is typically defined by connecting a set of nodes, in two or three-dimensional space, with line segments, yielding a finite set of structural members. It may be defined to include all possible node connections or only a subset of them (e.g. by connecting a node only with its n closest neighbors).

The optimization problem is then to find the subset of the structural members that minimizes, or maximizes, a physical quantity of the design. Moreover, the flexibility of the parameterization can be increased by allowing the design to have variable node locations (i.e. shape optimization) ([Imai and Schmit, 1981](#)), or by allowing the structural members to have variable cross-sections (i.e. structural sizing) ([Goldberg and Samtani, 1986](#)).

A major challenge of the method is that significantly large number of structural member combinations, i.e. 2^N where N is the total number of structural members, can be drawn from the ground structure. However, including only a small number of structural members in the ground structure may not capture the optimal, or even near-optimal, design from the structural universe. [Gilbert and Tyas \(2003\)](#) and [Pritchard et al. \(2005\)](#) mitigated the computational issue by using linear programming with the column generation technique, in order to iteratively add structural members with high ‘virtual strains’ to the optimization problem. Their method is capable of finding the optimal design from ground structures consisting of more than 100 million structural members. Linear programming is an efficient optimization method for types of optimization problems, such as mass minimization subject to stress constraints or compliance minimization subject to a volume constraint.

Considering such a large number of candidate structural members, the optimized design may also have geometrical complexity not suitable for practical purposes. Several approaches have been proposed in the literature in order to reduce the complexity of the optimized design by either addressing the complexity already in the problem formulation or applying a postprocessing step to the optimized design. [He and Gilbert \(2015\)](#) studied an algorithm, first proposed by [Prager \(1974\)](#) and [Parkes \(1975\)](#), where short structural members are made less favorable by penalizing all structural members by a fixed additional length during the volume evaluation. They conclude that the method is efficient, as it requires only a minor modification in the objective function, but sometimes fails to effectively simplify the structure. Later, [Torii et al. \(2016\)](#) noted that this occur in their experiments especially when structural members have similar lengths. [He](#)

and Gilbert (2015) studied also a postprocessing technique where the nodal coordinates of the optimized design are used as new design variables, and nodes moving close to each other are merged. Torii et al. (2016) and Asadpoure et al. (2015) proposed penalty function approaches, indirectly penalizing the number of active structural members or nodes in the design, in order to reduce the complexity of the final design. Gao et al. (2017) proposed a technique to improve the quality of the ground structure by placing its nodal points at the intersections of principal stress trajectories⁸. The technique can be used to generate well-defined ground structures with fewer structural members.

The truss topology optimization problems can also be tackled using evolutionary, or other population-based algorithms. The main benefits of evolutionary algorithms are that 1) they are robust optimization methods even for problems with complex physics-related objective and/or constraint functions, and 2) they can be used to simultaneously evolve the cross-sectional areas of the trusses, their topology and node locations.

Grierson and Pak (1993) were the first to apply a GA to simultaneous sizing, topology and shape optimization. Later, Rajan (1995) improved the efficiency of the approach, as well as its applicability to design of practical skeletal structures. In these early papers, the existence of structural members is encoded using binary variables and the cross-sectional areas and nodal coordinates using separate real-encoded variables, which, according to Deb and Gulati (2001), makes the representation sensitive to the binary variables. Deb and Gulati (2001) relate the existence of structural members to their cross-sectional area, by excluding structural members with small cross-sectional area from the structure. As a result, the whole representation is encoded using real numbers.

This type of approach is referred to as a *single-level optimization technique*. The argument against using *multi-level optimization techniques*, in which the three types of optimization are conducted in sequence, is that they may not always yield the global optimum as the optimization problems are not linearly separable. However, Luh and Lin (2008, 2011) showed that their two separate two-level optimization techniques based on ant colony optimization (Dorigo and Gambardella, 1997) and particle swarm optimization (Eberhart and Kennedy, 1995) yield, on several test cases, designs that are superior to those obtained by Deb and Gulati (2001).

Recently, a variety of different population-based algorithms have been applied to truss topology optimization problems. Examples of these algorithms are the cuckoo search algorithm (Yang and Deb, 2009), firefly algorithm (Yang and Deb, 2009) and teaching-learning based optimization (Rao et al., 2011), the applications of which are reported in the papers by Gandomi et al. (2013), Miguel et al. (2013) and Savsani et al. (2016), respectively. Kaveh and Zolghadr (2014) compared the performance of nine population-based algorithms on truss topology optimization problems, and defined a diversity index

⁸They determine the principal stress trajectories by solving an equivalent static optimization problem, the design domain of which is filled with homogeneous isotropic material.

to describe the trade-off between exploration and exploitation in these algorithms. They conclude that, according to their experiments, the best performing algorithms have a large diversity index at the beginning of the optimization process (i.e. exploration is favored), which then decreases gradually as the optimization proceeds.

2.5 Evolutionary structural optimization (ESO)

ESO is a structural optimization heuristic, which iteratively removes material, or structural members, with lesser utilization. The method was first presented by [Xie and Steven \(1992\)](#). Before starting an optimization process, the design space is discretized into finite elements of material, and each element is assigned a binary design variable, describing its existence in the structure. An iterative process is initiated from the full structure, wherein all existence variables are set to one. At each iteration, structural analysis is performed, and, based on the result, a sensitivity number is determined for all elements. The sensitivity number α_i represents the utilization level of the i th element in the structure. To improve the performance of the structure, elements with lesser utilization are rejected if their sensitivity number is less than the rejection sensitivity number

$$\alpha^{\text{rej}} = RR_j \alpha^{\text{max}}, \quad (2.4)$$

where RR_j is the prevailing rejection rate of the iteration j and α^{max} is the maximum sensitivity number in the structure at the same iteration j . At the beginning of the process, the rejection rate RR_j is given a low value, which is then increased during the process. Iterations are repeated until a predefined stopping criterion is met. The stopping criterion may be, for example, a minimum sensitivity number level for all elements in the structure or a desired volume fraction of the design space.

ESO has gained widespread popularity among researchers and practitioners, which has resulted in well over 100 published papers ([Huang and Xie, 2010b](#), [Munk et al., 2015](#)). As an optimization method, ESO is considered as a combination of intuitive-heuristic and gradient-based methods ([Eschenauer and Olhoff, 2001](#)). In contrast to SIMP, no intermediate material is present in the optimized structure.

However, the final structure is still prone to *checkerboarding* and *staircasing*. Checkerboarding means an occurrence of material and void elements in the design domain in a pattern of a checkerboard, which complicates the interpretation and manufacturing of the optimized structure. Checkerboarding can be avoided by filtering the design sensitivities ([Díaz and Sigmund, 1995](#)). In staircasing, the boundary of the structure has a shape of a staircase caused by the discrete element grid. An approach to obtain smooth boundaries is to use a nodal ESO (NESO) ([Chen et al., 2002](#)), in which material is added and removed by moving the boundary nodes of an unstructured finite element mesh.

This section provides a brief overview of the aspects of ESO that are relevant to the current work. For more information, the reader may wish to consult the ESO textbook by [Huang and Xie \(2010a\)](#) or the review paper on evolutionary topology optimization algorithms by [Munk et al. \(2015\)](#).

2.5.1 Sensitivity numbers

Structural optimization problems are often constrained by one or multiple physical quantities, such as stress, stiffness and natural frequency. In the ESO method, these constraints are followed by using a relevant sensitivity number. However, the sensitivity numbers of the elements do not explicitly constrain the design. Instead, when material with a low sensitivity number is removed, the optimization process is driven towards the optimal structure, with respect to the prevailing material volume. The process is terminated when the optimal design with the prevailing material volume is no longer sufficient to fulfill the constraints.

In early papers the utilization level of a material element was determined based on its von Mises stress σ_{vM} . In an ideal distribution of material, the von Mises stress distribution is constant throughout the structure. Thus, the sensitivity number of an element was defined to be the maximum von Mises stress in the element as

$$\alpha_i = \sigma_{vM}^{\max}. \quad (2.5)$$

Besides stress, stiffness is an important factor in many engineering applications. The stiffness constraint is included in the ESO method via the mean compliance, i.e. the inverse of stiffness, of the structure. The sensitivity number for the mean compliance is defined as

$$\alpha_i = \frac{1}{2} \mathbf{u}_i^T \mathbf{K}_i \mathbf{u}_i, \quad (2.6)$$

where $\mathbf{u}_i$ and $\mathbf{K}_i$ are the displacement vector and the stiffness matrix, respectively, of the i th element in the structure (c.f. [Chu et al. \(1996\)](#) for a derivation). By removing the elements with the smallest sensitivity to the mean compliance, the optimization process is driven towards the stiffest structure, with respect to the prevailing material volume. Although the sensitivity numbers are fundamentally different in stress and stiffness constrained optimization, [Li et al. \(1999b\)](#) showed that optimization runs with both constraints result in similar topologies.

In addition to the aforementioned sensitivity numbers, a wide range of other sensitivity numbers have been derived for optimization problems including multiple load cases, multiple materials, a design dependent-gravity load or constraints for displacement, natural frequency or linear buckling ([Huang and Xie, 2010a](#), [Munk et al., 2015](#)). In this study, one of our applications is topology optimization of the aircraft wing internal structure. Aircraft wings, being thin-walled structures, are prone to buckling. Therefore,

in the following, we examine the sensitivity number for critical buckling load in more detail.

The work on the sensitivity number for linear buckling is mainly due to [Manickarajah et al. \(1998, 2000\)](#). The governing equation for linear buckling of an elastic structure is

$$(\mathbf{K} + \lambda_j \mathbf{K}_g) \mathbf{u}_j = 0, \quad (2.7)$$

where $\mathbf{K}$ is the global stiffness matrix, $\mathbf{K}_g$ is the global geometric stiffness matrix, or the stress stiffness matrix, λ_j is the j th eigenvalue of the system and $\mathbf{u}_j$ is the corresponding eigenvector. Starting from the governing equation, [Manickarajah et al. \(1998, 2000\)](#) derived the sensitivity of the j th eigenvalue, i.e. the buckling load, to be

$$\Delta \lambda_j = \frac{-\mathbf{u}_j^T ([\Delta \mathbf{K}] + \lambda_j [\Delta \mathbf{K}_g]) \mathbf{u}_j}{\mathbf{u}_j^T [\mathbf{K}_g] \mathbf{u}_j}. \quad (2.8)$$

By normalizing the eigenvector ($\mathbf{u}_j^T [\mathbf{K}_g] \mathbf{u}_j = 1$), the equation simplifies to

$$\Delta \lambda_j = -\mathbf{u}_j^T ([\Delta \mathbf{K}] + \lambda_j [\Delta \mathbf{K}_g]) \mathbf{u}_j. \quad (2.9)$$

Two simplifications can be made to Equation 2.9. First, if the modification to the thickness distribution of the structure is sufficiently small, $\Delta \mathbf{K}_g$ may be neglected ([Manickarajah et al., 1998](#)). Second, considering a small change in the cross-sectional area or thickness in the i th element, the change in global stiffness matrix $[\Delta \mathbf{K}]$ is equal to the change in the local stiffness matrix $[\Delta \mathbf{k}_i]$ of the i th element. Therefore, Equation 2.9 is further simplified to

$$\Delta \lambda_{ij} = -\mathbf{u}_{ij}^T [\Delta \mathbf{k}_i] \mathbf{u}_{ij}, \quad (2.10)$$

where $\mathbf{u}_{ij}$ is the j th eigenvector of the i th element. Typically, optimization is constrained by the first eigenvalue ($j = 1$), which corresponds to the critical buckling mode. Thus, the sensitivity number of the i th element for linear buckling is

$$\alpha_i = \Delta \lambda_{i1} = -\mathbf{u}_{i1}^T [\Delta \mathbf{k}_i] \mathbf{u}_{i1}. \quad (2.11)$$

The sensitivity number may be used either to constrain the critical buckling load and minimize the structural mass, or to constrain the material volume of the structure and maximize the critical buckling load. In both cases, a limited amount of material is removed or redistributed from/in the structure. Two sensitivity numbers, α_i^{inc} and α_i^{dec} , are required for each element in the structure. These sensitivity numbers correspond to the increase and decrease, respectively, in the size variable A (thickness or cross-sectional area) of the element i . Considering a discrete step size ΔA in the size variable, local

stiffness matrices are

$$\begin{cases} [\Delta \mathbf{k}_i]^{\text{inc}} = \mathbf{k}_i(A + \Delta A) - \mathbf{k}_i(A) \\ [\Delta \mathbf{k}_i]^{\text{dec}} = \mathbf{k}_i(A - \Delta A) - \mathbf{k}_i(A). \end{cases} \quad (2.12)$$

$$(2.13)$$

The two sensitivity numbers α_i^{inc} and α_i^{dec} are obtained by substituting Equations 2.12 and 2.13 into Equation 2.11.

A challenge in topology optimization with the buckling constraint is that the structure nearly always has multiple eigenvalues. A structural modification that increases the first eigenvalue might at the same time decrease the second eigenvalue, which then becomes the critical buckling mode. Manickarajah et al. (1998, 2000) presented a solution where the sensitivity number is the average of the first n eigenvalues. Later, Rong et al. (2001) improved the method for closely-spaced and repeated eigenvalues.

The derivation of the sensitivity number for buckling assumes that the topology of the structure remains the same, and only small modifications are made to the size variables of the structure. Therefore, the method is not applicable in this form to topology optimization with the ground structure approach, wherein wholesale changes of structural components are intended.

2.5.2 Bi-directional ESO (BESO)

In the ESO method, the rejection of an element is irreversible, which may cause an optimization process to convergence prematurely to a local optimum. To overcome this problem, Querin et al. (1998) introduced bi-directional ESO (BESO), where rejected elements may be recovered to the structure. In addition, the BESO method enables an optimization run to be initiated from an intermediate design containing both solid and void elements. The approach reduces the number of required iterations in comparison to the ESO method, initiated from a significantly over-designed structure.

The challenge in the BESO method is in the determination of sensitivity numbers for void elements. Since these elements are not included in the latest iteration, their sensitivity numbers can only be determined implicitly. After the paper by Querin et al. (1998), several improvements have been made to the implicit estimation of void sensitivities. Two studies from the early development are worth mentioning. Yang et al. (1999) used linear extrapolation of the displacement distribution in the solid elements to estimate the displacements in the void elements. The authors applied the method to a stiffness maximization problem. Querin et al. (2000) defined a von Mises stress criterion, where solid elements with the lowest von Mises stress are rejected and void elements near the highest stress regions are recovered. The authors demonstrate that even if the optimization is initiated from the minimum amount of material to transfer the load (regardless of the stress level) the results are the same as with the ESO method. Later, Huang and Xie (2007) developed a filtering scheme for node-based sensitivity numbers,

where an element sensitivity number is defined to be a weighted average of sensitivity numbers of the nodes laying closer than d^{limit} from the center of the element. The closer the node is to the center of the element, the more weight it has on the element sensitivity. The filtering scheme not only enables the accurate determination of void sensitivities but also overcomes two limitations of the ESO method: the checkerboarding and mesh dependency (Huang and Xie, 2010a).

Finally, let us consider an example of applying the BESO method to topology optimization. The objective of the optimization is to maximize the stiffness of a cantilever beam, which design domain has a rectangular shape. A downward point load is applied to the center point of the right-hand side boundary. The left-hand side boundary is clamped, whereas all the other boundaries are free. The material volume is constrained to 50% of the total design domain. Starting from the initial guess (Figure 2.3(a)), material is iteratively rejected and recovered, yielding the optimized structure in Figure 2.3(b). The main feature of BESO, recovering elements back to the structure, is clearly visible in two subfigures.

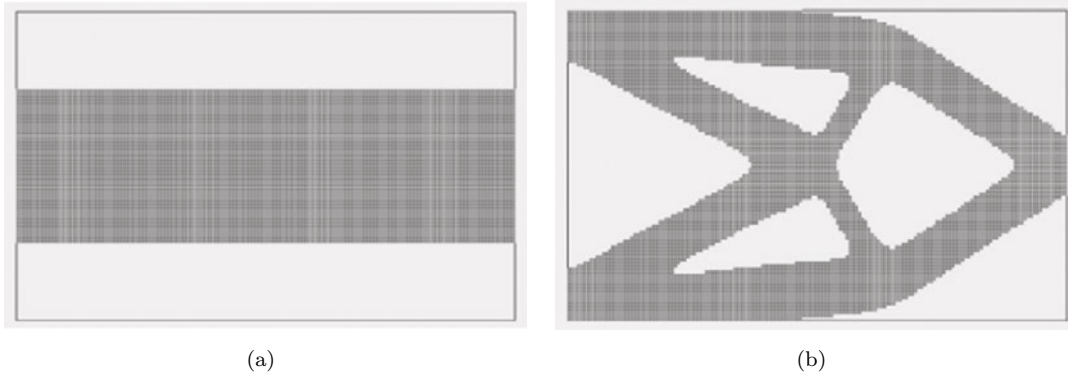


Figure 2.3: Volume-constrained stiffness maximization of a cantilever structure using BESO (Huang and Xie, 2010a). The initial guess (a) and the optimized structure (b) are presented.

2.6 Other topology optimization methods

Two other widely used topology optimization methods, namely Solid Isotropic Material with Penalization (SIMP) and the Level-Set Method (LSM), are worth mentioning in order to paint a clearer picture of the broader context. As the main focus of this work is on evolutionary optimization methods, these methods are not studied later in this work. However, the SIMP method will be used as a benchmark in Chapter 6. The methods are described briefly in the following two sections. A reader not interested in these topics may wish to move to Section 2.7 without loss of continuity.

2.6.1 Solid Isotropic Material with Penalization (SIMP)

SIMP is a topology optimization method, first introduced by [Bendsøe \(1989\)](#), which uses the continuum material definition (see Section 2.2). Since its introduction, the method has been extensively used and developed. SIMP is a gradient based method that exploits the adjoint method in its design sensitivity analysis. Its advantage in comparison to other topology optimization methods, especially non-gradient methods, is its effectiveness in terms of the computational cost.

In the method, each material element x in a design domain Ω is assigned a scalar design variable $\bar{\rho}(x) \in [0, 1]$, representing its normalized density. The actual density ρ of the material element x is linearly proportional to the design variable, so that if $\bar{\rho}(x) = 0$ the material element is a void, and if $\bar{\rho}(x) = 1$ the density of the material element is equal to a reference density ρ_0 .

The method considers isotropic material only. Assuming Poisson's ratio ν to be independent of the normalized density, Young's modulus of a material element, $E(x)$, is defined to be proportional to the normalized density as

$$E(x) = \bar{\rho}(x)^r E_0, \quad (2.14)$$

where E_0 is the actual Young's modulus of the material and $r > 1$ is a penalty coefficient ([Bendsøe and Sigmund, 2003](#)). The greater the penalty coefficient r is, the more intermediate densities are penalized. A penalty coefficient of $r \geq 3$ is usually needed to for a clear 0/1 material distribution ([Bendsøe and Sigmund, 2003](#)). Typically, the total volume of the material is constrained as

$$\int_{\Omega} \bar{\rho}(x) d\Omega \leq \phi, \quad (2.15)$$

where ϕ is the desired volume fraction.

The simplest, yet the most studied, optimization problem with SIMP is to minimize the compliance of a structure subject to a volume constraint. Moreover, the method has been applied to optimization problems with a variety of objective and constraint functions. For an extensive listing of the types of applications, the reader may wish to consult the review papers by [Rozvany \(2009\)](#) and [Deaton and Grandhi \(2014\)](#).

Figure 2.4 presents an example topology optimization result obtained by SIMP, where the objective is to minimize the compliance of a short cantilever beam. The design domain is defined to be a rectangular area, where the left-hand side boundary is fixed and other boundaries are free. A point load is applied to the bottom right corner of the domain. The figure illustrates the occurrence of intermediate densities (see the grey regions in Figure 2.4). The intermediate densities could be eliminated from the solution by further increasing the penalty coefficient r .

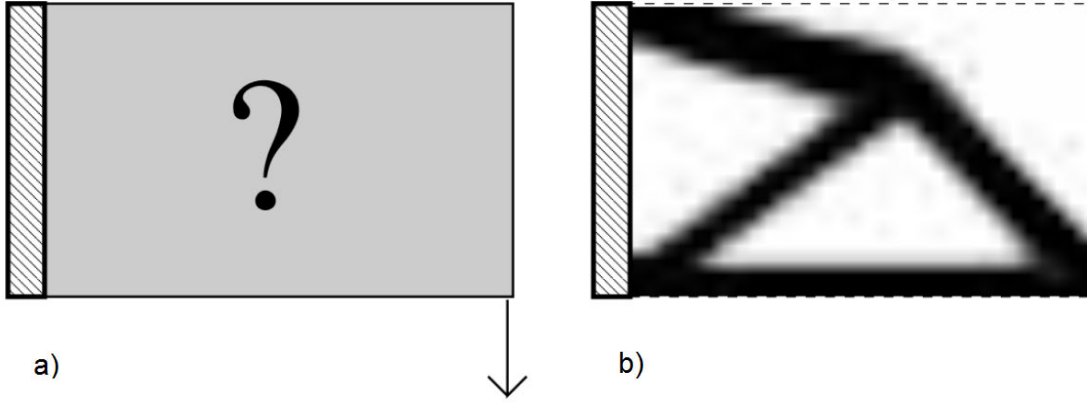


Figure 2.4: Topology optimization of a short cantilever beam using the SIMP method (Sigmund, 2001).

The method has the following disadvantages, which are also typical for the most topology optimization methods with the continuum material definition. First, an undesirable checkerboarding phenomenon, already encountered in Section 2.5, might occur in the final solution. Second, SIMP is considered mesh-dependent, meaning that the same optimization problem with two different mesh discretizations may yield two significantly different results. Third, being a gradient-based optimization method, SIMP might not yield the global minimum. The probability of obtaining the global minimum can be improved by increasing the penalty coefficient in increments during the optimization process (Sigmund and Petersson, 1998).

2.6.2 Level-set method (LSM)

The LSM is a topology optimization method, where the optimal material-void boundaries, or the boundaries between two materials, are sought implicitly via iso-contours of a scalar level-set function ϕ_{ls} . The introduction of the LSM is due to Osher and Sethian (1988), who developed the method in order to model moving boundaries. Sethian and Wiegmann (2000) presented the first application of the method to topology optimization.

Considering a design comprising of only a single material, the design domain Ω includes the material domain M and the void domain $\Omega \setminus M$. Using the level-set function ϕ_{ls} , the design is commonly defined as

$$\begin{cases} \phi_{ls}(x) > c_b \Leftrightarrow x \in M & (2.16) \\ \phi_{ls}(x) = c_b \Leftrightarrow x \in \Gamma & (2.17) \\ \phi_{ls}(x) < c_b \Leftrightarrow x \in (\Omega \setminus M), & (2.18) \end{cases}$$

where x is a point in the design space, Γ is the material boundary of the design and c_b is a scalar constant, usually defined to be 0 (van Dijk et al., 2013). To visualize the use of

the LSM, an optimization process of a two-dimensional cantilever beam is presented in Figure 2.5, along with the corresponding level-set functions in Figure 2.6. The objective of the optimization process is to minimize the mean compliance of the structure subject to 50% volume constraint.

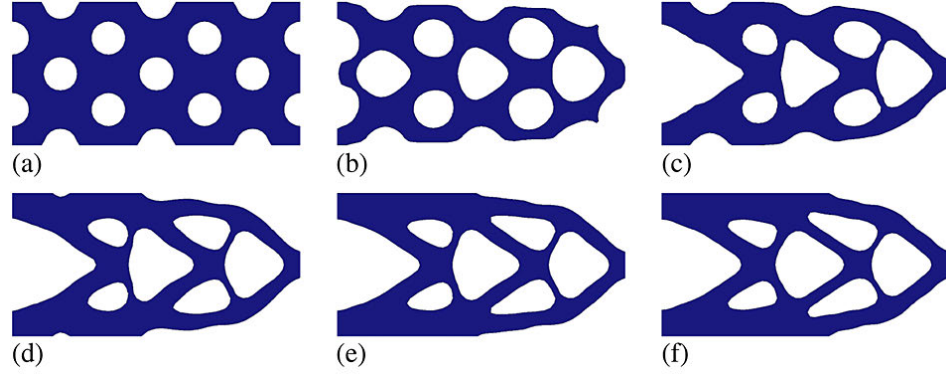


Figure 2.5: The development of the structural topology in a meshless level-set optimization process (Luo et al., 2012). The structural topology is plotted at generations 1 (Subfigure a), 25 (b), 75 (c), 150 (d), 300 (e) and 481 (f).

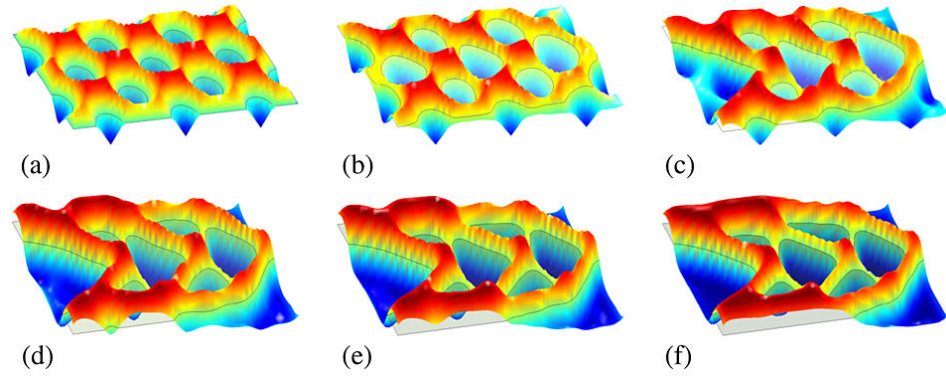


Figure 2.6: Level-set functions corresponding to the structural topologies presented in Figure 2.5 (Luo et al., 2012).

An iso-contour, extracted from the level-set function, is mapped into a mechanical model either using a discrete definition of the void-material boundary (referred as conforming discretization), immersed boundary techniques or a density-based approach (van Dijk et al., 2013). The conforming discretization requires either remeshing of the FE model at every iteration or using a meshless structural analysis method. Immersed boundary techniques retain the majority of the element mesh but apply local enrichments near the void-material boundaries. In the density-based approach, a predefined element mesh is used and the elements at the material boundary Γ are assigned a density linearly proportional to the material fraction of the element, c.f. the SIMP method (Section 2.6.1) with $r = 1$.

The LSM has several advantages and disadvantages in comparison to methods using the continuum material definition. The biggest advantage of the LSM is that the optimized

structures have crisp boundaries, and, therefore, the structure is free from undesired features seen with other topology optimization methods, such as intermediate material (SIMP, Section 2.6.1) and staircasing (ESO, Section 2.5). However, this only applies when either conforming discretization and immersed boundary techniques are applied to the mapping step. The material boundaries with the density based approach are blurred. Disadvantages of the LSM are a tendency to converge to a local minimum and dependency on the initial guess. For more information on the LSM, the reader may wish to consult the review paper by [van Dijk et al. \(2013\)](#).

2.7 Lindenmayer systems (L-systems)

L-systems were introduced by [Lindenmayer \(1968a,b\)](#), who studied the developmental process of multicellular organisms, in the late 1960s. The fundamental idea of L-systems is that complex objects (e.g. plants) can be modeled by repeatedly modifying a simple object by following a set of predefined rewriting rules. The number of times the rewriting rules are applied represents the age of the organism. In addition to biology, L-systems have been applied to a variety of other fields, such as computer graphics, artificial intelligence and engineering.

The language of rewriting rules is referred to as *formal grammars* ([Chomsky, 1956](#)), or *Chomsky grammars*. Several types of grammars, or systems, are derived from the work of Chomsky, such as L-systems and *shape grammars* ([Stiny, 1975](#)). Techniques based on the latter have been applied in several architecture and urban area modeling applications. All these grammars are initiated from a starting string or shape, and the rewriting rules are applied iteratively until a termination criterion is fulfilled. A single developmental stage in both L-systems and shape grammars may be performed in parallel or in sequence, while it may only be performed in sequence in Chomsky grammars ([Prusinkiewicz and Lindenmayer, 2012](#)). The rewriting rules in L-systems are applied to the string that represent the graph, whereas in shape grammars they are applied directly on the geometric features of the graph ([Prusinkiewicz, 1986](#)).

Using the taxonomy of L-systems, the process is started from a (simple) initial object, called the *axiom* ω_0 . Further, the state of the system after the rewriting rules are applied n times is referred to as its n th *developmental stage* ω_n . Both the axiom and rewriting rules are defined using an alphabet Σ of letters and/or symbols, which are referred to as *characters*. The left and right-hand sides of a rewriting rule are referred to as the *predecessor* and *successor*, respectively.

Let us consider a simple example⁹, where the alphabet $\Sigma \equiv [a, b]$, the axiom $\omega_0 = b$ and rewriting rules are $P_1 : a \rightarrow ab$ and $P_2 : b \rightarrow a$. To obtain the first developmental stage,

⁹This example is, in fact, equivalent to formal grammars, but we describe it here using the taxonomy of L-systems.

the axiom letter ‘ b ’ is converted into ‘ a ’ due to the rewriting rule P_2 , and therefore $\omega_1 = a$. When the rules are applied further, the following developmental stages are obtained: $\omega_2 = ab$, $\omega_3 = aba$, $\omega_4 = abaab$, $\omega_5 = abaababa, \dots$

In L-systems, these sequences of characters are interpreted into graphs that represent living organisms. In the following, two interpretation formalisms, the *turtle interpretation* and *map L-systems*, are presented through examples. In addition, they are reviewed as parameterization methods for topology optimization. For an extensive review of L-systems, the reader may wish to consult the text book by [Prusinkiewicz and Lindenmayer \(2012\)](#).

2.7.1 Turtle interpretation

In the turtle interpretation, the sequences of characters are interpreted into geometries via a moving turtle (cf. the turtle feature in the programming language LOGO). The orientation of the turtle is defined by its axial coordinates and heading. Each letter or symbol in the sequence is a command for the turtle, such as ‘move ahead by distance d ’ or ‘turn clockwise by angle $\Delta\theta$ ’. The moving turtle draws the lines of the geometry while executing the series of commands.

This section demonstrates the development of an example plant using L-systems and the turtle interpretation. The example is presented by [Prusinkiewicz and Lindenmayer \(2012\)](#). Let us consider an alphabet Σ containing the letters F and X , and symbols ‘+’, ‘-’, ‘[’ and ‘]’. Our example plant is defined by the following input:

$$\begin{aligned} \text{Axiom:} \quad & \omega_0 = X \\ \text{Rules:} \quad & P_1 : F \rightarrow FF \\ & P_2 : X \rightarrow F[+X][-X]FX \\ \text{Parameters:} \quad & \Delta\theta = 25.7^\circ \end{aligned} \tag{2.19}$$

The process is started by generating the character sequence of the desired developmental stage, in the same way as in the previous example. Following the axiom ω_0 , the next two developmental stages of the system are:

$$\omega_1 = F[+X][-X]FX \tag{2.20}$$

and

$$\omega_2 = FF[+F[+X][-X]FX][-F[+X][-X]FX]FFF[+X][-X]FX. \tag{2.21}$$

Next, the sequences are translated into geometries using the turtle interpretation. The characters have the following meaning for the turtle:

- letters move the turtle forward by step size d (the moving turtle draws a line, having the width w),
- symbol '+' turns the turtle anti-clockwise by angle $\Delta\theta$,
- symbol '-' turns the turtle clockwise by angle $\Delta\theta$,
- symbol '[' commands the turtle to stack its orientation, and
- symbol ']' returns the turtle to the previously stacked orientation.

The last two symbols enable the creation of branches, as the turtle may return to a previously visited location. The axiom ($n = 0$) and the first three developmental stages of the system ($n = 1 \dots 3$) are interpreted into plants in Figure 2.7.

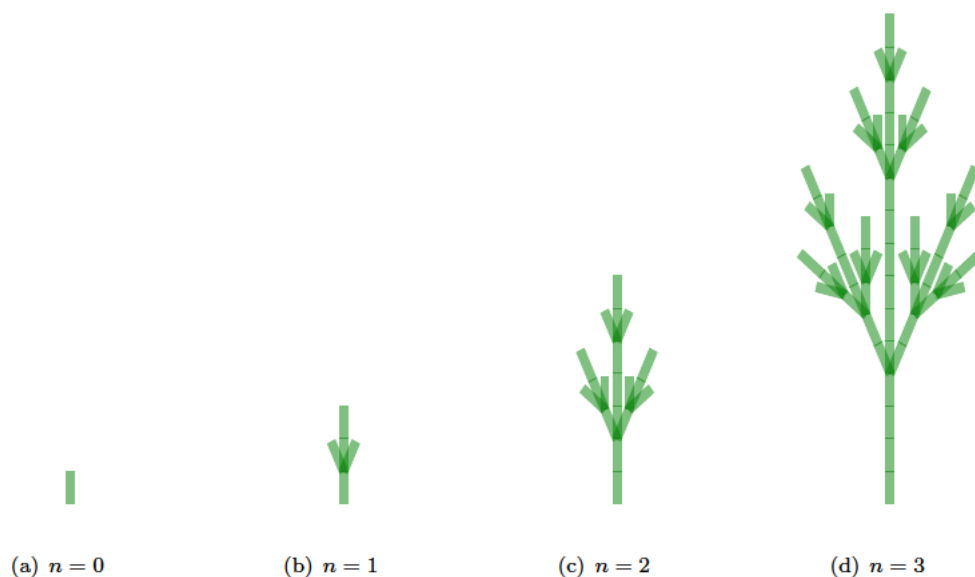


Figure 2.7: Visualization of a growing plant, interpreted from the L-system defined in Equation 2.19. The variable n is ordinal of the developmental stage. Thus, $n = 0$ corresponds to the axiom ω_0 , and $n = 1 \dots 3$ to the three subsequent developmental stages of the plant.

2.7.1.1 Turtle interpretation of parametric symbols

Using the above described turtle commands, L-systems are restricted to produce geometries consisting of line segments with lengths that are integer multiples of the step size d , and further, the angles between the line segments are restricted to be integer multiples of the turning angle $\Delta\theta$.

More complex geometries, free of these restrictions, can be generated by parametric L-systems (Prusinkiewicz and Hanan, 1990), in which symbols are associated with numerical values. In this work, we use the following parametric symbols:

- $\$(\theta)$ turns the turtle by angle θ (positive direction being anti-clockwise),
- $@(c_s)$ changes the prevailing step size to $d_i = d_{i-1}c_s$, and
- $\&(c_w)$ changes the prevailing line width to $w_i = w_{i-1}c_w$.

Symbols '[' and ']' stack and unstack the prevailing attributes, step size d_i and line width w_i , in the same way as the orientation. To demonstrate the use of parametric symbols, let us consider another example L-system, which is defined by the following input:

$$\begin{aligned} \text{Axiom: } \omega_0 &= X \\ \text{Rules: } P_1 &: F \rightarrow FF \\ P_2 &: X \rightarrow F\&(\frac{2}{3})[\$(\frac{\pi}{4})X][\$(\frac{\pi}{12})X]F@(\frac{2}{3})X \end{aligned} \quad (2.22)$$

Again, following the axiom ω_0 and the second rewriting rule P_2 , the first developmental stage ω_1 becomes

$$\omega_1 = F\&(\frac{2}{3})[\$(\frac{\pi}{4})X][\$(\frac{\pi}{12})X]F@(\frac{2}{3})X. \quad (2.23)$$

Graphical interpretations of the first four developmental stages of the L-system are shown in Figure 2.8 (see the topological congruence to Figure 2.7). While the topology of the plant is defined based on the order of the characters in the axiom and the rewriting rules, the parameters associated with the symbols define its shape.

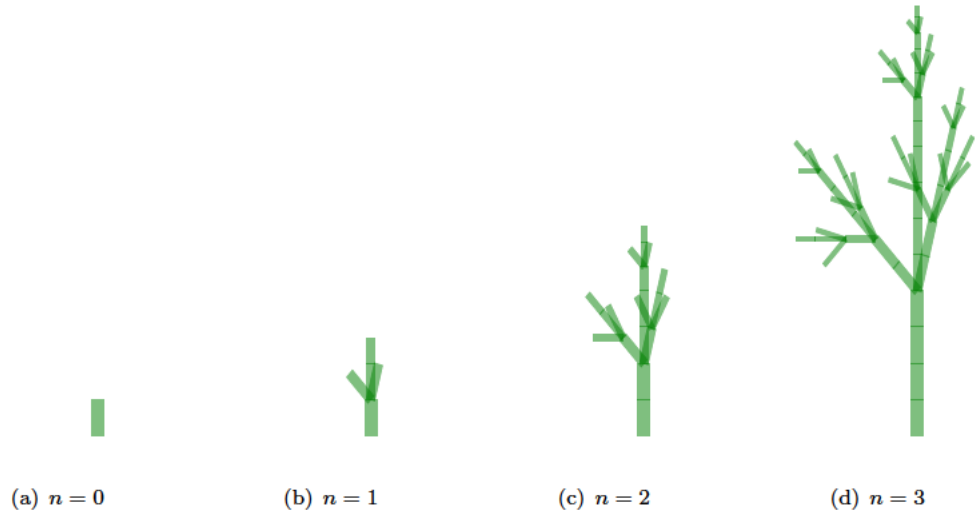


Figure 2.8: Visualization of the parametric L-system defined in Equation 2.22.

In order to better associate the interpreted graphical features with the parametric symbols, let us further examine Equation 2.23 and Figure 2.8(b), which both correspond to the first developmental stage ω_1 of the parametric L-system. The written format starts by letter 'F', which the turtle interprets as a step forward; this step forms the root trunk of the plant. The parametric symbol ' $\&(\frac{2}{3})$ ' commands the turtle to change the

prevailing line width to a fraction of $2/3$ of the previous line width. Next, the sequence $['(\frac{\pi}{4})X']$ forms the left branch of the plant by commanding the turtle to 1) stack its orientation, prevailing line width and prevailing step size, 2) turn left by angle $\pi/4$, 3) take a step forward, and 4) unstack the orientation and prevailing properties. Similarly, the sequence $['(-\frac{\pi}{12})X']$ forms the right branch of the plant. Finally, the remaining sequence $['F@(\frac{2}{3})X']$ forms the center branch of the plant by commanding the turtle to 1) take a steps forward, 2) change the prevailing step size to a fraction of $2/3$ of the previous step size, and 3) take another step forward.

2.7.2 Map L-systems

While experimenting with the early formalisms of L-systems, scientists understood that the method worked well on simple path-like structures, but was not capable of modeling the development of more complex geometries in botany, such as cellular layers. For that purpose, [Lindenmayer and Rozenberg \(1978\)](#) developed more general systems, enabling the formation of cycles that consist of a finite set of edges (e.g. cells in an organism). The system was later refined by [Nakamura et al. \(1986\)](#).

In these systems, referred to as the map L-systems, the rewriting is done in two phases. First, the rewriting rules divide edges and introduce markers. Contrary to the turtle interpretation with brackets, no branches are allowed to form at this point. They act as start and end points for new edges, which split cells¹⁰. In terms of the notation, a start of a marker is indicated by the symbol '[' and the end of a marker by the symbol ']'. The inside of a marker contains two characters: a symbol '+' or '-' allocating the side of the marker (left or right) and a letter referred to as the label (see the first rewriting rule P_1 of the example map L-system in Equation 2.24). Second, marker pairs are searched inside each cycle and matching markers are connected by a new edge. For two markers to be considered matching, they must be located inside the same cycle and have the same label. In the case of multiple matching marker pairs, only the first found is connected¹¹ ([Prusinkiewicz and Lindenmayer, 2012](#)).

Let us consider an example, where the alphabet consists of letters A , B and z , and the axiom and the rewriting rules are defined as

$$\begin{aligned} \text{Axiom: } \omega_0 &= ABAB \\ \text{Rules: } P_1 : A &\rightarrow B[-A]z[+A]B \\ P_2 : B &\rightarrow A \end{aligned} \tag{2.24}$$

¹⁰Markers have a counterpart in biology, preprophase bands of microtubes ([Prusinkiewicz and Lindenmayer, 2012](#)).

¹¹However, usually when the method is applied to topology optimization, additional criteria are included, such as a minimum fraction of the offspring cycle area in comparison to the parent cycle area, or a minimum angle between a new edge and the surrounding edges. If criteria are not fulfilled, the marker pair is ignored. See for example the paper by [Pedro and Kobayashi \(2011\)](#).

The letter z is a terminal letter, for which no rewriting rule is assigned. In this example, the axiom represents a unit square containing only a single cycle $ABAB$, where each letter represents an edge in the unit square ($n = 0$ in Figure 2.9). The edges are ordered clockwise starting from the bottom edge. By terms *letter* and *edge*, we essentially refer to the same substructure of a map L-system, but a letter corresponds to a written substructure and an edge to a drawn substructure. When moving to the next developmental stage, all letters are rewritten, and edges redrawn, based on the rewriting rules.

Let us consider either of the two A edges in the axiom ($n = 0$). According to the rewriting rule P_1 , the letter A is rewritten into $B[-A]z[+A]B$, which, in terms of the redrawing, means that the edge A is split into edges B , z and B , all having an equal length, and markers $[-A]$ and $[+A]$ are added to the edge nodes according to their location in the rewriting rule. To finalize the developmental stage, markers with the same label are searched inside the cell, and the first matching marker pair, fulfilling predefined criteria, is connected with a new edge ($n = 1$ in Figure 2.9). In this example, the only criterion is that all offspring cycles must have a non-zero area. In a written format, the first developmental stage is

$$\omega_1 = Bz[A]BABzBA, \quad (2.25)$$

where the square bracketed A is the new edge that proceeds from the bottom of the unit square to the top. In this notation, the square brackets indicate a branch in the same way as in the string L-systems. Finally, the remaining markers are discarded. The subsequent developmental stages are generated by repeating the same process. Figure 2.9 visualizes the first four developmental stages of the system.

More precisely, this type of interpretation formalism is referred to as *Binary Propagating Map OL-systems with markers* (mBPMOL-systems) (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. In the remainder of this work, we will refer to the mBPMOL-systems simply as map L-systems.

Later in this study we use directional markers. Possible directions for the markers are ‘ $\leftarrow$ ’, ‘ $\rightarrow$ ’ 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 ‘ $\rightarrow$ ’ or neutral, and the direction of the end marker ‘ $\leftarrow$ ’ or neutral.

Map L-systems can be extended by a dynamic method in order to obtain even better biological relevance to the cellular layers (Prusinkiewicz and Lindenmayer, 2012). This involves applying an osmotic pressure inside each cell, and assuming that the edges will remain straight and follow Hooke’s law in their axial direction. The osmotic pressure

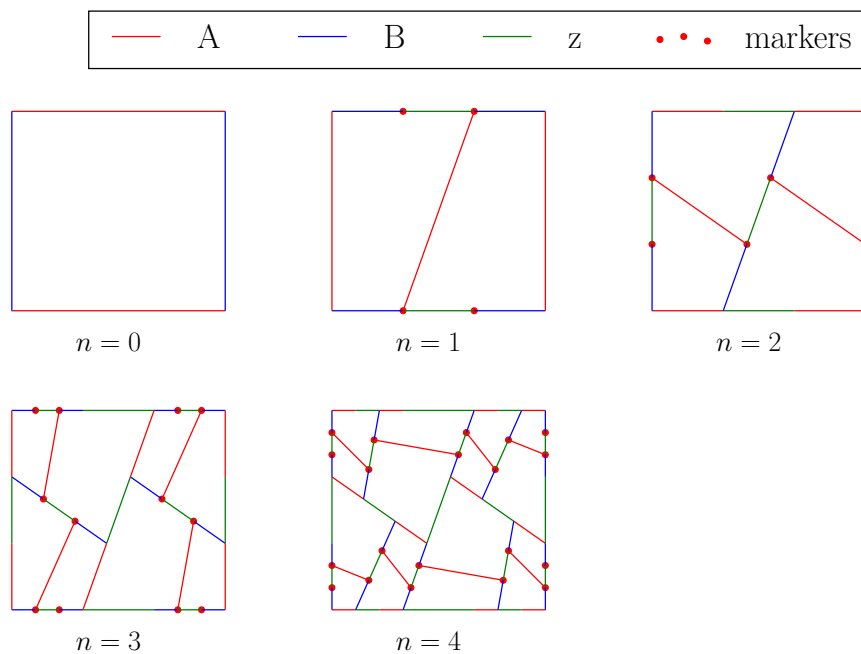


Figure 2.9: Visualization of a cellular division method. The graph shows the axiom ($n = 0$) and first four developmental stages ($n = 1 \dots 4$) of a map L-system (redrawn from the text book by [Prusinkiewicz and Lindenmayer \(2012\)](#)).

on an edge is linearly proportional to the edge length and inversely proportional to the area of the cell. The equilibrium of the vertex locations is solved at every developmental stage.

The developmental stages presented in Figures 2.7-2.9 are only a few example topologies that may be generated by L-systems. A diverse range of different topologies may be generated by varying the axiom and the rewriting rules, and further by including more letters in the alphabet. In the next section, we describe how topologies are evolved using a GA.

2.7.3 Optimization via a GA

The majority of the L-systems-based topology optimization studies in the literature use GAs, which we reviewed in Section 2.3, to seek the optimal distribution of material. A key factor for a well-performing algorithm is how the formalism of L-systems is encoded into a genetic representation, suitable for a GA. A numerical representation for L-systems with the turtle interpretation is described for example by [Kobayashi \(2010\)](#). In the following, we describe a popular numerical representation for map L-systems by [Pedro and Kobayashi \(2011\)](#).

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

$$\mathbf{x} = \left[\underbrace{x_{a,1} \ x_{a,2} \ \dots \ x_{a,N_a}}_{\text{Axiom } \omega_0} \underbrace{P_1 \ P_2 \ \dots \ P_{N_P}}_{\text{Rewriting rules } P_j} \underbrace{x_1 \ x_2 \ \dots \ x_{N_v}}_{\text{Additional variables}} \right]. \quad (2.26)$$

Each letter of the axiom, having a total of N_a letters, is represented as a real number $x_{a,i}$. 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 $\Sigma \equiv \{A, B, C\}$, the real number $x_{a,i}$ 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, excluding the terminal letter z if it exists in the alphabet¹². Each rewriting rule P_j is encoded into N_r sets of real numbers, called *tokens*, as

$$P_j = \left[\beta_{j,1} \ \beta_{j,2} \ \dots \ \beta_{j,N_r} \right]. \quad (2.27)$$

A token is a part of the right-hand side of a rewriting rule and may appear as a 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 right-hand side of a rewriting rule. The k th token of the j th rewriting rule is encoded into a set of six real numbers as

$$\beta_{j,k} = \left[x_1 \ x_2 \ x_3 \ x_4 \ x_5 \ x_6 \right], \quad (2.28)$$

where the real numbers encode the token as follows:

1. Existence of the token: an empty token if x_1 in $[0, p_{\text{empty}}]$, else token exists. If the token is empty, the real numbers $x_2, \dots, x_6$ are ignored.
2. Letter: A if x_2 in $[0, \frac{1}{N_a}]$, else B if x_2 in $[\frac{1}{N_a}, \frac{2}{N_a}]$, else C if x_2 in $[\frac{2}{N_a}, \frac{3}{N_a}]$, $\dots$
3. Orientation: ' $\rightarrow$ ' if x_5 in $[0, \frac{1}{3}]$, else 'neutral' if x_5 in $[\frac{1}{3}, \frac{2}{3}]$, else ' $\leftarrow$ '.
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_3 in $[0, \frac{1}{2}]$, else the side is '-'.
6. Edge property: '/' if x_6 in $[0, \frac{1}{3}]$, else 'neutral' if x_6 in $[\frac{1}{3}, \frac{2}{3}]$, else '*'.

Pedro and Kobayashi (2011) define the last real number of a token, the edge property, to decrease, '/', retain, 'neutral', or increase, '*', a specific property (e.g. the thickness) of the offspring edge by a quantum amount in comparison to the parent edge. In their

¹²A terminal letter appearing in the system will remain the same for all subsequent developmental stages.

application, this variable is used to control the variation of the thickness distribution in a cantilever structure.

Finally, a total of N_v additional variables are encoded into the genotype. The additional variables always contain the age of the system, i.e. the ordinal of the developmental stage, 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 the new cycles. These requirements can define, for example, a minimum angle between two edges belonging to a cycle, a minimum fraction for the area of an offspring cycle in comparison to the parent cycle, or a 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

$$n_d = N_a + 6N_rN_p + N_v. \quad (2.29)$$

2.7.4 Applying L-systems to topology optimization

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 converged 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 (Figure 2.10). 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. [Li et al. \(2018\)](#) implemented a global-local topology optimization method, in which the outer loop is the map L-systems-based method driven by a GA and the inner loop is based on the LSM. The

authors demonstrate the method on multiple compliance minimization problems of two-dimensional structures¹³. Their method yields lower compliances on two problems studied by [Pedro and Kobayashi \(2011\)](#), using the map L-systems-based method, by margins of 12.86% and 28.29%.

[Allison et al. \(2013\)](#) and [Khetan et al. \(2015\)](#) studied topology optimization of truss structures using the map L-systems-based method, to which they included a nested sizing loop using sequential linear programming. The authors ensured the structures to be mechanisms-free by accepting only cell divisions that result in two triangular cells. [Sabbatini et al. \(2015\)](#) applied L-systems, with turtle interpretation, to multi-objective stiffer layout optimization, minimizing the vibration amplitude and mass of a plate structure.

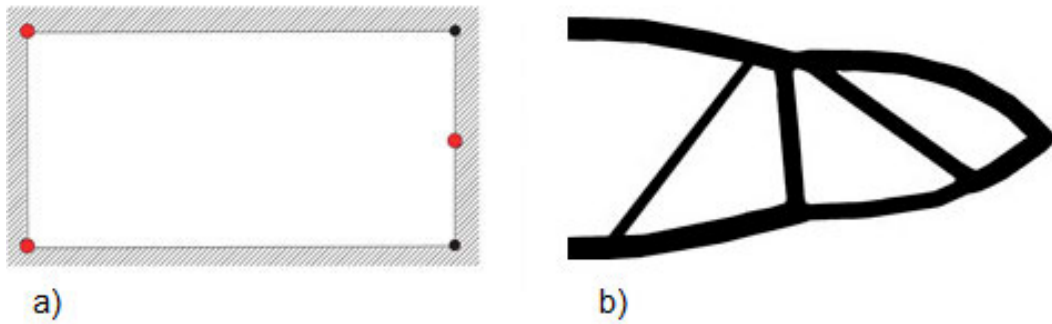


Figure 2.10: Stress-constrained mass minimization of a cantilever structure with the map L-system-based parameterization ([Pedro and Kobayashi, 2011](#)). The design space is a rectangular area (a), where the lower- and uppermost points on the left-hand side boundary are fixed and a load is applied on the center point on the right-hand side boundary. The optimized structure (b) has a volume that is 26% of the total the design space volume.

The first application to the topology optimization of an aircraft wing internal structure is due to [Kobayashi et al. \(2009\)](#), who demonstrated the suitability of the map L-systems-based parameterization to structural optimization of a generic fighter aircraft wing box (the work was later extended by [Kolonay and Kobayashi \(2010\)](#) to include aerodynamic shape parameters, as well as panel buckling and flutter constraints). Later, [Kolonay and Kobayashi \(2015\)](#) studied the weight- and L/D -optimal fighter aircraft wing shapes and topologies using map L-systems as the parameterization method. They adopted a bilevel optimization algorithm, where the higher level considers design variables related to topology, shape and control surfaces, and structural sizing is performed on the lower level. [Stanford et al. \(2012\)](#) used the map L-systems in an optimization study of a flapping wing venation topology. They present Pareto fronts of optimized venation patterns between thrust generation, lift generation and input power requirements. In a

¹³As well as on a multi-objective optimization problem of simultaneously minimizing the compliance and volume of the structure.

subsequent paper (Stanford et al., 2013), the authors also included the topology of the flapping mechanism in the optimization.

As we can see from the previous two paragraphs, 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, 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.

Finally, in the studies that use map L-systems, a common practice is to parameterize the age of the system, i.e. the number of times the rewriting rules are applied, as one of the additional variables. However, the number of cells in the geometry description increases, at most, quadratically as a function of the age variable (cf. doubling time of cells in biology). Therefore, a minor change in the age variable may cause a major change in the phenotype, which is not beneficial for the evolvability of the algorithm. Here, we propose that an approach, avoiding this problem, could be to parameterize the number of cell divisions (instead the age of the system). Such an approach has not been reported in the literature. Let us refer to this approach as the *linearization of the age variable*.

2.8 Selected areas of application

In this work, we have selected three areas of application, to which the studied evolutionary topology optimization methods are applied. Following the aims of the research (Section 1.1), the areas are selected based on two main aspects. First, the three areas as a whole represent a diverse set of physics-based objective and constraint functions, in which direct and generative encodings may be compared. Second, each of them comprise interesting engineering design problems, and thus the developed algorithms, and obtained results, have the potential to be useful for practitioners working in these areas.

A literature review of these areas is presented in the following. The areas are aircraft wing structures (Section 2.8.1), conductive heat transfer systems (Section 2.8.2) and integrally stiffened panels (Section 2.8.3).

2.8.1 Aircraft wing structures

The Wright brothers constructed their first aircraft wings using wood and fabric, with a structural arrangement consisting of longitudinal spars and transverse ribs. During World War I aircraft engineers introduced cantilever wings free of external struts or wires that cause unnecessary drag. The skin material was changed from fabric to wooden veneer, which enabled so-called *stressed skin* designs, leading to a reduction in wing mass (Jakab, 1999). For the first time, spars, ribs, and stressed skins formed the load carrying wing box structure. Further, longitudinal stiffeners were attached to the skin panels of the wing box to prevent buckling.

This structural arrangement is still widely used in aircraft design, regardless of the growing range of new materials. However, it is questionable if the arrangement of longitudinal spars and stiffeners, and transverse ribs is optimal in terms of the structural mass of the wing, especially if the wing has an unconventional outer mold line (OML) shape.

This section reviews topology optimization applications to the aircraft wing internal structure. First, we examine the applications in terms of their design space parameterization and, second, in terms of their optimization methodology. Aircraft wings are typically slender structures, which are prone to buckling. Therefore, third, we review the handling of buckling constraints in topology optimization. Finally, we provide a short review of available parametric geometry frameworks for aircraft components.

2.8.1.1 Design space parameterization

The design region of the aircraft wing internal structure is limited by two important functionalities of the wing. First, the internal structure must be located inside the outer mold line (OML) of the wing¹⁴. Any external structural member would obviously disturb the aerodynamic characteristics of the wing. Second, the wing encapsulates fuel tanks, high lift devices and other systems that may represent internal boundaries for the design region.

In the literature, both continuum and discrete material definitions have been applied to topology optimization of the aircraft wing internal structure. We review relevant studies in the following. Figures 2.11 and 2.12 show representative examples of optimized structures obtained in these studies.

Starting from the continuum material definition, Maute and Allen (2004) searched for the optimal topology of an aeroelastic structure, where minimum required aerodynamic performance was given as a constraint. Stanford and Ifju (2009) maximized the L/D ratio of a micro air vehicle wing via aeroelastic analysis. James and Martins (2012) (Figure 2.11(a)) and Dunning et al. (2014) used continuous, unstructured three-dimensional

¹⁴We do not consider externally braced wings here.

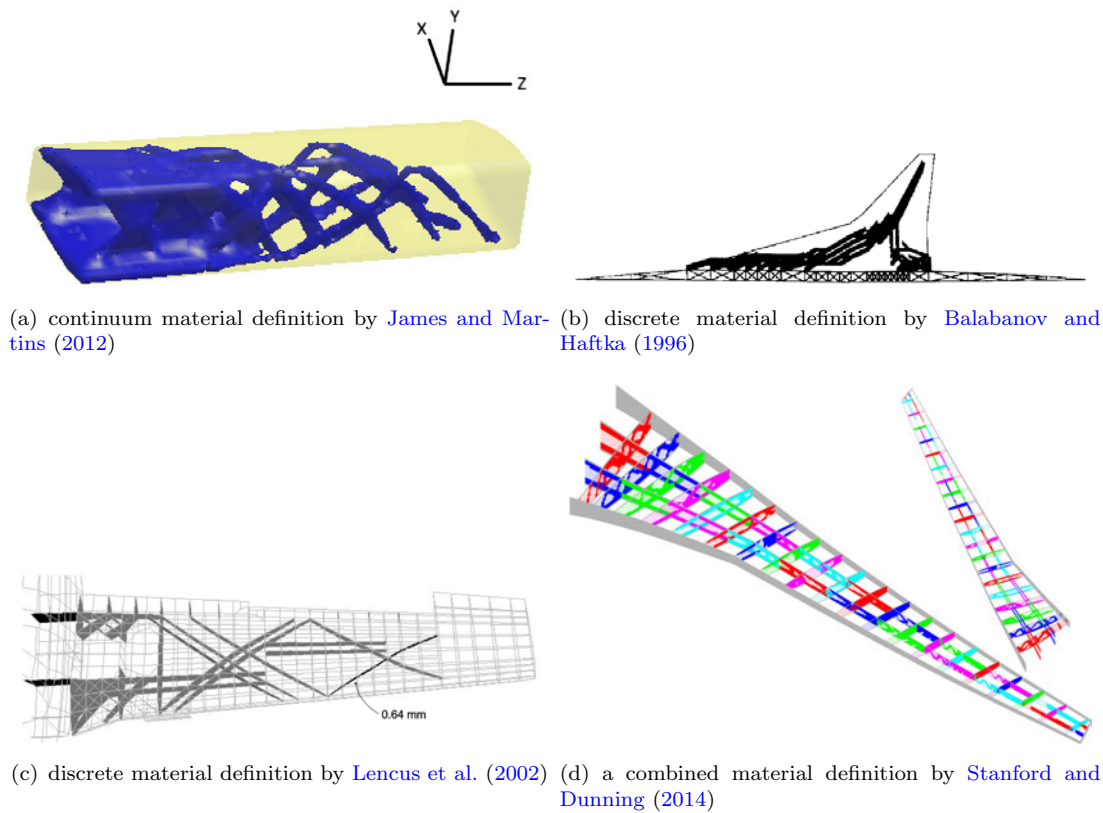


Figure 2.11: Continuum and discrete material definitions in topology optimization studies on aircraft wing structures.

grids to minimize the compliance of a wing structure. Oktay et al. (2014) used the homogenized material definition in a topology optimization process that was coupled with Computational Fluid Dynamics (CFD)-based aerodynamic load analysis. Eves et al. (2009) also used initially the continuum material definition and interpreted the result into shell-type structural members. Several other authors, such as Eschenauer and Olhoff (2001) and Krog et al. (2004), used the continuum material definition in a two-dimensional domain to seek the optimal material distribution of wing box ribs.

Recently, Aage et al. (2017) used the continuum material definition (with the SIMP method) with an outstanding design resolution to minimize the compliance of the NASA Common Research Model (Vassberg et al., 2008) wing (Figure 2.12). Their parameterization consists of over one billion design variables, and thus the final design has 200 times finer resolution than the previous state-of-the-art techniques (Langelaar, 2017). The authors estimate that the obtained design would result in 2-5% reduction in the wing mass in comparison to a conventional internal structure¹⁵. However, the large number of design variables requires an enormous computational resource of 1-5 days on a high performance computing cluster of 8000 Central Processing Units (CPUs) – a resource not generally available in routine design practice.

¹⁵Subject to the availability of a large-scale additive manufacturing technique in the future.

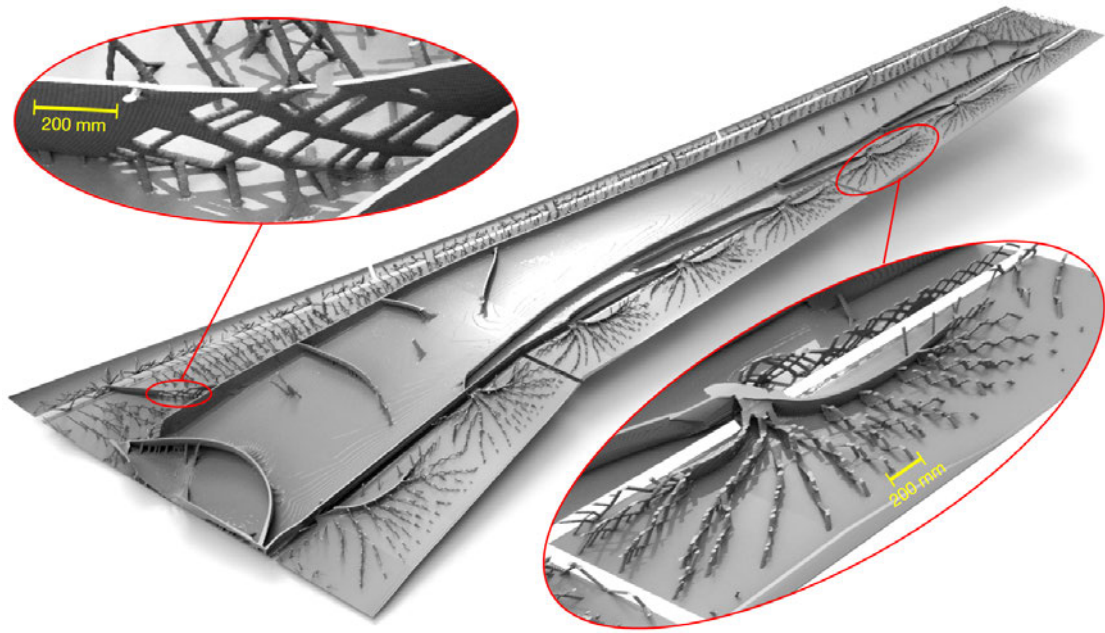


Figure 2.12: Optimized design, minimizing the compliance of the NASA Common Research Model wing, obtained by the SIMP method (Aage et al., 2017).

Several studies have also been conducted using the discrete material definition. Balabanov and Haftka (1996) applied the ground structure approach with beam elements to structural optimization of the high speed transport aircraft wing (Figure 2.11(b)). The members of the ground structure were oriented in multiple directions. In the ground structure approach by Wang et al. (2011), the structural members are oriented in spanwise direction of the wing and may appear as full-depth spars or as partial-depth stiffeners. Similarly, Yang et al. (2016) used a ground structure of spanwise oriented structural members in the wing box, but also included multi-directional structural members in the landing gear region. Lencus et al. (2002) used a ground structure consisting of spanwise, chordwise and diagonal structural members, with an area (clear of structure) reserved for the landing gear. Snyder and Weisshaar (2014) searched the optimal topology of the wing internal structure for a multiple flying configurations via the ground structure approach.

Maute and Allen (2004) and Stanford and Dunning (2014) (Figure 2.11(d)) used a combination of continuum and discrete material definitions. First, they seeded the wing box by discrete structural members, and, second, apply the continuum material definition to each of the structural members. In the optimization procedure, they vary the material distribution of the structural members while keeping their locations fixed.

Stanford and Dunning (2014) point out that the topology optimization studies on aircraft wings using the continuum material definition, such as those by James and Martins (2012) and Dunning et al. (2014), do not yield spar-rib like structures. They name two obvious alternative reasons for this. The first is that the physics of the model, its

load cases or boundary conditions, or the constraints of the design variables were not implemented correctly in the optimization studies. The second is that the obtained non-traditional internal structures offer better performance than the traditional spar-rib structure. If the latter reason is true, their observation is very important in the conservative field of aircraft structural design, where the traditional rib-spar topology has been used for over a century.

The results with the continuum material definition include highly three-dimensional shapes that cannot be manufactured using sheet-type materials. In practice, most of the shapes in the internal structure of the wing are constrained by manufacturability. While the structures of small unmanned air vehicles (sUAVs) could be manufactured by additive manufacturing methods, the internal structure of a large passenger aircraft must be assembled using metallic or composite sheets. The ground structure approach with shell and beam members is likely to produce internal structures that have better manufacturability, because only instances of sheet- and stiffener-like structures with a constant thickness or constant profile dimensions, respectively, are allowed to emerge in the structure.

Several studies indicate that the addition of diagonally oriented structural members as a part of the internal structure improves the total performance of the structure. This behavior is seen in the studies with curvilinear spars and ribs by [Locatelli et al. \(2011\)](#) and [Jutte et al. \(2014\)](#). [Lencus et al. \(2002\)](#) present results of the ground structure approach that also include many diagonal structural members. In addition, [Eves et al. \(2009\)](#) interpret optimization results obtained by continuum material definition to sheet-type structural members, resulting in an internal structure with several diagonally oriented structural members.

In the introduction (Chapter 1), we described generative encodings being suitable for describing geometries with self-similar and hierarchical sub-parts. The design obtained by [Aage et al. \(2017\)](#) (Figure 2.12) represent an optimized structure with by far the finest available resolution. Despite being obtained by a direct encoding method, the design has a noticeable number of repeating patterns of structural members. The trailing edge spar is a bifurcating load bearing structure, which initiate from the root of the wing and spreads into six to eight similar flower-like patterns comprising of full-depth and truss structural members. The entire leading edge is supported by similar patterns of truss structural members. [Aage et al. \(2017\)](#) conclude that this pattern resembles the cellular structure seen natural bones, such as in the beak of the hornbill bird. These self-similar features obtained by the high resolution direct encoding method are encouraging results to further develop generative encoding methods.

2.8.1.2 Optimization methods

Design optimization of the aircraft wing is one of the major fields where topology optimization has been applied. This section presents relevant studies where gradient- and/or non-gradient-based optimization methods have been applied to aircraft wing design. In addition, some review papers comparing gradient- and/or non-gradient-based topology optimization methods are discussed.

At least the following gradient-based optimization methods have been implemented in topology optimization of an aircraft wing: SIMP (Maute and Allen, 2004, Eves et al., 2009, Stanford and Ifju, 2009, Oktay et al., 2014, Stanford and Dunning, 2014, Aage et al., 2017), ESO (Lencus et al., 2002, Kelly et al., 2014) and the LSM (James and Martins, 2012, Brampton et al., 2012, Dunning et al., 2014). Contrary to other papers with SIMP, Stanford and Dunning (2014) searched the optimal material distribution in pre-seeded two-dimensional spars and ribs inside the wing, whereas others include the entire volume inside the wing as the three-dimensional design domain. The objective function of previously mentioned papers, excluding papers by Maute and Allen (2004) and Stanford and Ifju (2009), is to minimize the compliance of the structure subject to a volume constraint. Eves et al. (2009) minimized afterwards the mass of the structure subject to twist angle, von Mises stress and buckling constraints. However, at this point only the rib pitch and material thicknesses were used as design variables. Lencus et al. (2002) checked the feasibility of the structure afterwards in terms of buckling and wing tip deflection constraints, but these constraints were not considered in the optimization process.

Non-gradient-based methods have also been used in the topology optimization of the aircraft wing internal structure. However, to improve efficiency, non-gradient-based methods are often coupled with gradient-based methods. While gradient-based methods are typically applied to the continuum material definition, non-gradient-based methods are mostly applied to a discrete ground structure.

Wang et al. (2011) applied a non-gradient-based method, called *ant colony optimization* (Dorigo and Gambardella, 1997), along with the gradient-based structural optimizer MSC.Nastran Sol200, to seek the best combination of structural members from a given ground structure. Yang et al. (2016) used a combination of a GA and mixed integer optimization to find the best combination of ground structure members. The objective of the optimization was to minimize the structural mass subject to a range of constraints, including Tsai-Hill strength criteria and the lowest buckling mode. Snyder and Weisshaar (2014) propose a two-level optimization algorithm, in which elements with lesser utilization are removed using the *performance-based optimization* (Liang, 2005) on the higher level, and structural sizing is performed on the lower level. The study by Hansen and Horst (2008) is worth mentioning though it is not on the aircraft wing. In the study, a combination of an evolution strategy and the gradient-based MSC.Nastran

Sol200 optimizer are used to minimize the mass of a body section of the blended-wing-body aircraft. The optimization is constrained by buckling and either von Mises stress or Tsai-Hill failure criteria constraints, depending on the selected material.

We reviewed the L-systems-based topology optimization studies on aircraft wings (which also use non-gradient-based methods) already in Section 2.7.4. In these studies (Kobayashi et al., 2009, Kolonay and Kobayashi, 2010, 2015), the available structural member types are limited to full-depth structural members, although wings with effective use of material are typically constructed of both full-depth and stiffener-like structural members.

2.8.1.3 Buckling constraints

Aircraft wings are constructed of lightweight and slender structural members, which, under a compressive load, are prone to buckling. Thus, the critical buckling load is to be considered early in the design process, to ensure the feasibility of the design and avoid (potentially expensive) design changes later in the process. In terms of optimization, the critical buckling load is generally treated as a lower limit constraint. However, there are also examples in the literature where the critical buckling load is the objective function (of the maximization type), and the design space is constrained by a volume or mass constraint.

The evaluation of the critical buckling load differs for gradient- and non-gradient-based topology optimization methods. Gradient-based methods require design sensitivities of the function, whereas non-gradient-based methods generally penalize designs that invalidate constraints to keep the design feasible (see Section 2.3.3 for the use of penalty functions in GAs). The latter is often easier to implement because no gradient information is needed. On the other hand, gradient-based methods have the potential to make the optimization process more effective. The design sensitivities in gradient-based methods can be determined for example by a range of finite difference methods or by the adjoint method. The adjoint method is computationally more efficient than finite difference methods, especially if the optimization problem has many design variables (Keane and Nair, 2005).

Several topology optimization papers with the continuum material definition have been published where the lowest critical buckling load is either a constraint or the objective function. For example, Neves et al. (1995), Pedersen (2000) and Sekimoto and Noguchi (2001) included buckling evaluation in SIMP (reviewed earlier in Section 2.6.1), whereas Manickarajah et al. (1998, 2000) and Rong et al. (2001) included buckling evaluation in ESO (reviewed earlier in Section 2.5). The buckling evaluation with the continuum material definition, e.g. SIMP, is problematic due to the occurrence of spurious non-physical local buckling modes in areas of low density (Neves et al., 1995, 2002). The spurious localized modes can be eliminated by removing elements with a low density

from the topology using an appropriate threshold value (Tenek and Hagiwara, 1994). However, this manipulation may cause the optimization process to oscillate and, therefore, interrupt its convergence (Bendsøe and Sigmund, 2003). Recently, Browne et al. (2012) presented a binary programming method that is free of spurious buckling modes. The method uses the analytic first-order derivatives of the stress stiffness matrix with respect to the densities of the topology.

These studies are conducted mainly for simple two-dimensional structures. Very few papers in the literature consider buckling as a constraint in aircraft wing topology optimization and evaluate it using a finite element (FE) model of the whole wing structure. One rare exception is the work by Yang et al. (2016), wherein the buckling constraint is included as a penalty function in a GA-based optimization procedure. Some authors, such as Lencus et al. (2002), mention buckling as a constraint but do not evaluate it during the optimization process. Instead, they check the buckling constraint after the optimization process. To our knowledge, the more advanced methods of including buckling constraints, presented in the previous paragraph, have not been implemented into the topology optimization of the whole aircraft wing.

2.8.1.4 Parametric geometries

Parametric geometries are an essential building block for any geometry optimization framework. They are a means of mapping a geometry into a finite set of design variables that are varied in the optimization process. The requirements for a parametric geometry are conciseness, robustness and flexibility (Sóbester, 2014). The parameterization must be very carefully chosen. Too rigid a parameterization can lead to a limited design space that does not include the optimal design. On the other hand, the risk in a flexible parameterization is the so-called ‘curse of dimensionality’ and its increased demand for computational resources. Several approaches to implement parametric geometries have been presented, such as parametrization with discrete points, polynomials or B-splines (Sóbester and Forrester, 2014).

Kulfan and Bussolletti (2006) presented a method, called class-shape transformation (CST), for describing an aircraft geometry through a set of class and shape functions. Class functions define the type of the geometry, e.g. an airfoil- or a body-type cross-section, whereas shape functions define the description of the shape. The CST can be performed to build the OML of the main aircraft components: a lifting surface, body and engine nacelle. The first is built in Cartesian coordinates, whereas the last two are built in axisymmetric coordinates.

Several parametric geometry frameworks have been developed for conceptual design of aircraft. AirCONICS (Aircraft Configuration through Integrated Cross-disciplinary

Scripting) (Sóbester and Forrester, 2014, Sóbester, 2015) is a library of NURBS (Non-Uniform Rational B-Spline) objects to define aircraft OML shapes. It is build on light-weight computer-aided design software Rhinoceros 3D via its Python scripting interface. The library is designed to be a suitable tool for the whole aircraft design process from preliminary to detailed design (Sóbester, 2015). GeoMACH (Hwang and Martins, 2012) is a parametric geometry module by the MDO Laboratory at the University of Michigan. The code, written in Fortran, defines separately the OML and the internal structure of an aircraft. GeoMACH belongs to a wider conceptual optimization framework, called OpenMDAO (Gray et al., 2010). GENAIR (Gagnon and Zingg, 2013) is another framework capable of providing parametric geometries for OpenMDAO. According to Gagnon and Zingg (2013), it provides geometries with better quality near a junction between two components than GeoMACH, but has a higher computational cost to define the junction geometries. OpenVSP (Hahn, 2010) is a framework developed in California Polytechnic State University primarily for aircraft OML shape generation, but it also includes tools for generating the main internal structure (i.e spars, ribs and bulkheads). It is based on an earlier framework referred to as Rapid Aircraft Modeler (RAM) (GlouDEMANS et al., 1996). OpenVSP includes a library of more than a hundred OML shapes of aircraft. SUAVE (Lukaczyk et al., 2015) is conceptual design environment to model and analyze unconventional aircraft configurations. All these frameworks are open source.

The number of design variables representing a design is critical especially in optimization procedures where the objective function evaluation is computationally expensive. Multi-level parameterization schemes consist of hierarchical levels, each of which have a specific number of design variables. These schemes can be used to vary the fidelity of the parameterization, or to apply multiple (nested) optimization algorithms to the same optimization process.

One such parameterization is based on subdivision surfaces, often used in computer graphics, which can represent a smooth surface via a coarse polygon mesh. The smooth surface is generated by recursively dividing its edges using specific division rules. Masters et al. (2017a) showed that a multi-level subdivision parameterization, where the number of design variables is periodically increased, outperforms the equivalent single-level method on nine tested two-dimensional optimization problems. The authors also applied the method to aerodynamic optimization of a three-dimensional wing (Masters et al., 2017b). The parameterization has some similarities to the L-systems-based methods, which we reviewed in Section 2.7. Both methods are extensively used in computer graphics because of their ability to represent complex geometries in a compact format. When used as a parameterization method, they both enable the definition of hierarchical design variables.

2.8.2 Conductive heat transfer systems

Engineering systems, such as electronic devices, generators and satellites, generate heat, which is to be efficiently transferred to the ambient environment to ensure their long life span, high performance and compact size. Optimal heat transfer capability often requires the heat exchanger components to have complex shapes and topologies. The recent development of additive manufacturing technologies, especially 3D printing of metals, have enabled the production of such components. Concurrently, various topology optimization methods have been presented to maximize the thermal efficiency of engineering systems, considering conduction, convection and conjugate heat transfer (Dbouk, 2017). Out of these three, conductive heat transfer problems clearly have the lowest computational cost, as they only require a numerical solution to a second order differential equation over the design domain. Therefore, in this work, we consider conductive problems as the first step towards more complex and realistic problems. In the reminder of this section, we provide a short review on topology optimization of conductive thermal systems. For an extensive review of the topic, as well as on topology optimization of other types of thermal systems, the reader may wish to consult the paper by Dbouk (2017).

The majority of the published papers tackling this topic consider steady-state conduction inside a rectangular, two-dimensional design domain. Bejan (1997) defined the so-called ‘volume-to-point’, or ‘area-to-point’, design problem where a finite design domain, with a uniformly distributed heat generation rate, is filled with high and low conductivity materials. The objective is to minimize the average or maximum temperature over the domain by distributing a limited amount of high conductivity material, to efficiently transfer the produced heat to the heat sink, which is a short section of the domain boundary. The remaining boundary conditions are adiabatic. This problem has been extensively studied and has become a popular benchmark in the field of thermal engineering.

To solve the problem, Bejan (1997) applied constructal theory, which is based on observations from the nature. According to this theory, the solutions are constructed from blocks with different designs and sizes, and, for each scale, their geometric details are determined theoretically to minimize their conductive resistance, which is a non-dimensional expression of their maximum temperature. Li et al. (1999a) modified ESO heuristics, initially developed for structural optimization, to be suitable for a conductive steady-state heat transfer problem. Gao et al. (2008) presented modified BESO heuristics for a similar optimization problem, and studied both design-dependent and independent heat load cases. In these studies, (B)ESO heuristics were applied to a design problem where the heat sink extends over the entire domain boundary. Later, Marck (2012) applied ESO to the problem defined by Bejan (1997). Cheng et al. (2003)

studied the problem using the *bionic optimization approach*, where the conductive material domain is iteratively expanded near regions where their temperature gradients are the highest and removed from regions where they are the smallest. The cellular automaton is another approach to the problem and its first application is due to [Boichot et al. \(2009\)](#). The algorithm aims at minimizing thermal gradients, or heat fluxes, at the boundary between high and low conductive materials. The authors, as well as [Marck \(2012\)](#), describe the method as being a simple way of obtaining a reasonable, tree-like solution, which, however, is likely to be sub-optimal.

According to the dedicated scientific literature, the most promising methods to solve this design problem are based on the density interpolation approach ([Bendsøe, 1989](#)). These methods were initially developed for structural topology optimization, where the discretization of the Partial Differential Equation (PDE) is typically conducted using the Finite Element Method (FEM). [Gersborg-Hansen et al. \(2006\)](#) were the first to obtain the design sensitivities from the Finite Volume Method (FVM), and used them in conjunction with topology optimization. [Marck et al. \(2012\)](#) used the SIMP method, with an aggregated objective function approach, in a multi-objective optimization study in order to minimize both average and variance temperatures over the design domain. [Dirker and Meyer \(2013\)](#) tested a variety of objective functions and problem related parameters of the SIMP method, and reported their results using non-dimensional measures for thermal conductivity and ‘definiteness’, i.e. how well intermediate material is eliminated from the final design. Their results show that the final design is highly dependent on the penalization coefficient value. [Dede \(2009\)](#) and [Burger et al. \(2013\)](#) applied the SIMP-based methodology to a three-dimensional volume-to-point design problem. The Method of Moving Asymptotes (MMA) ([Svanberg, 1987](#)) is used as the underlying gradient-based optimizer in all of the papers cited in this paragraph.

Apart from constructal theory, the aforementioned studies are based on gradient-based approaches. However, evolutionary algorithms have also been applied to this problem. [Xu et al. \(2007\)](#) used separately both Genetic Algorithms (GA) and simulated annealing ([Kirkpatrick et al., 1983](#)) to seek the optimal combination of discretized design domain elements. Later, [Boichot and Fan \(2016\)](#) show that their carefully tuned GA yields discrete designs having lower non-dimensional thermal resistances than studies using the cellular automaton ([Mathieu-Potvin and Gosselin, 2007](#), [Boichot et al., 2009](#), [Marck, 2012](#)), constructal theory ([Bejan, 1997](#), [Ghodoossi and Egrican, 2003](#)) and ESO ([Marck, 2012](#)), and similar thermal resistances to a study carrying out the SIMP approach ([Marck et al., 2012](#)). However, their algorithm requires an order of five million function evaluations to reach full convergence.

Faster convergences with evolutionary algorithms may be obtained by using alternative strategies to parameterize the design space. [Pedro et al. \(2008\)](#) parameterized the geometry of a tree-like structure (by parameters defining branch angles and lengths), and used a GA to minimize the maximum temperature of the design domain. They

showed that the level of geometric complexity has a considerable effect on the optimized objective function value. The approach by Lohan et al. (2017) is to use a generative encoding, based on the *space colonization algorithm* (Runions et al., 2005), in which the design space is first seeded with a set of attraction points, and then a branching structure is iteratively constructed to ‘colonize’ these points. Their objective is to minimize the thermal compliance of the design domain, which is discretized using both structured and unstructured meshes. Guo et al. (2018) proposed a generative encoding approach based on artificial neural networks. The approach uses a variational autoencoder (Kingma and Welling, 2013), the purpose of which is to reduce the dimensionality of the design via its latent layers, and deep convolutional neural networks (Krizhevsky et al., 2012), to prevent the appearance of disconnected high conductive material in the designs.

Nearly all of the above mentioned studies report that their optimized designs feature branching tree-structures, yet Dede (2009) describe his results to have self-similar features. L-systems and its turtle interpretation (Section 2.7.1) naturally produce geometries with branching tree-structure. However, as far as we are able to ascertain, the parameterization based on L-systems and its turtle interpretation has not directly been applied to the problem.

2.8.3 Integrally stiffened panels: natural frequency maximization

Integrally stiffened panels, made from metal or composites, are often seen in aircraft and satellite structures. Those made from metal are manufactured using methods like welding, casting or subtractive manufacturing techniques, such as face milling or chemical etching, while the corresponding composite panels are typically fabricated using a single layup and cure process. Thus, integrally stiffened panels compose of significantly fewer structural components than those manufactured by traditional techniques, such as riveting.

As the assembly of an aircraft requires more than 50% of its total manufacturing workload (Mei and Maropoulos, 2014), the reduced part count of integrally stiffened panels provides significant savings in the manufacturing cost. In addition, these panels are free of attachment flanges and holes, which enable lighter designs and longer life span in comparison to panels constructed via traditional manufacturing techniques (El-Soudani, 2006). The reason for the longer life span is the absence of attachment holes, which in traditional panels cause stress concentrations and, therefore, reduce the fatigue life of the component.

The stiffeners on the panel are typically oriented in two to four directions, resulting in repeating patterns of geometric shapes (Huybrechts et al., 2002). Two commonly used patterns consist of triangular (Figure 2.13(a)) and square shapes (Figure 2.13(b)), termed as *isogrid* and *orthogrid*, respectively. The prefix ‘iso’ for the triangular-patterned grid

denotes its isotropic stiffness properties in the plane of the panel. In comparison to the honeycomb sandwich construction, i.e. another commonly used panel structure, the grid structures have better damage tolerance and lower manufacturing costs ([Huybrechts et al., 1999](#)).

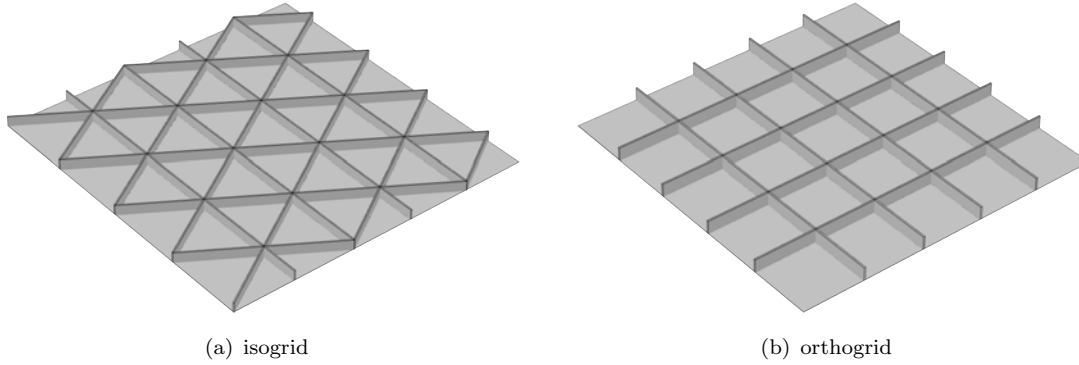


Figure 2.13: Integrally stiffened panels

As a part of a bigger assembly, an integrally stiffened panel is typically supported only at its edges. The wider the span of the panel is, the lower is its fundamental natural frequency, and thus the more prone it is to vibration at low excitation frequencies.

In the literature, various studies have been conducted on the methods of minimizing the vibration amplitude of grid structures, as well as other types of satellite structures, via damping or maximizing their fundamental natural frequencies¹⁶ via geometric choices. [Chen and Gibson \(2003\)](#) studied the use of passive viscoelastic damping layers embedded in composite isogrid structures in order to reduce their vibration amplitude. [Moshrefi-Torbati et al. \(2003\)](#) demonstrated a passive vibration control method for a satellite boom structure, consisting of a lattice of beam elements. In addition, several studies are conducted on active damping methods of reducing the sound penetration ([Gardonio and Elliott, 1999](#), [Yuan et al., 2015](#)) and vibration amplitude ([Beck et al., 2011](#)) of aluminum grid structures. [Akl et al. \(2008\)](#) determined the optimal stiffener angles for an isogrid panel in order to maximize its fundamental or first six natural frequencies.

When the design space of the panel stiffening is further relaxed, the design task becomes a topology optimization problem. The first application of topology optimization to maximize the fundamental natural frequency of a structure, using a methodology based on the homogenization method, is due to [Díaz and Kikuchi \(1992\)](#). Subsequently, the popular topology optimization methods, presented in this chapter, i.e. ESO (Section 2.5), SIMP (Section 2.6.1) and LSM (Section 2.6.2), have all been applied to vibration related design problems; their first applications are due to [Xie and Steven \(1994\)](#), [Pedersen \(2000\)](#) and [Osher and Santosa \(2001\)](#), respectively.

¹⁶A 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.

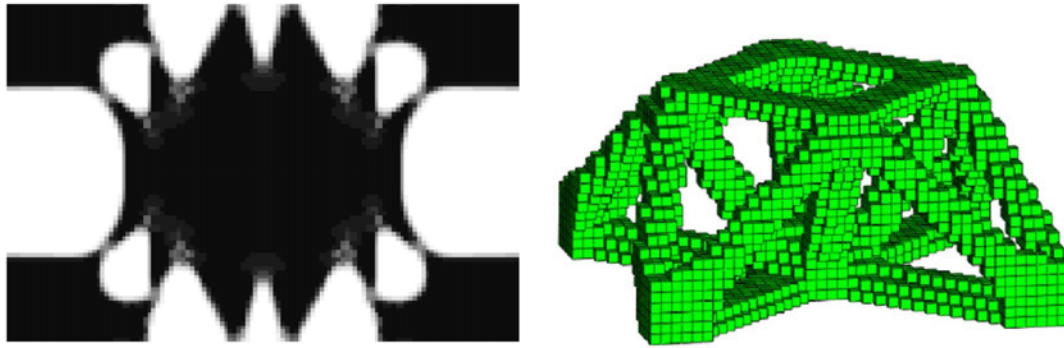
The most common objective of vibration related topology optimization studies is to maximize the fundamental natural frequency subject to a mass constraint. Alternatively, the objective may also be defined as to maximize the smallest difference between the natural frequencies and a predefined frequency (representing the excitation frequency), or to match the natural frequencies with a desired set of frequencies.

In the SIMP method, a key concern is in the treatment of low density regions, which may cause artificial, localized vibration modes (Neves et al., 1995, Pedersen, 2000). These modes have very low natural frequencies, and thus mislead the design process. The behavior is fundamentally the same as low density elements causing spurious buckling modes (see Section 2.8.1.3). Various approaches have been proposed to prevent the occurrence of localized vibration modes. Pedersen (2000) proposed a variation of the SIMP method where, for elements which density factor is less than 10%, the penalization of the stiffness is only one hundredth of the penalization of the mass. Tcherniak (2002) prevented the occurrence of localized modes by imposing the mass of low density elements to zero. Du and Olhoff (2007) used a large penalty coefficient of $r = 6$ for the mass of elements with a density factor less than 10% (while otherwise using a penalty coefficient of $r = 3$).

Another concern in vibration topology optimization with gradient-based methods is the changing order of natural frequency modes during the optimization process. It is important to trace the individual modes when using these methods, as, otherwise, the changing order of modes can cause design sensitivities to be discontinuous. Kim and Kim (2000) traced the modes using the modal assurance criterion, which is a statistical indicator describing the consistency between two mode shapes.

Let us next review some of the presented applications using gradient-based topology optimization methods; here, we focus on those that are relevant to the optimization problem of seeking the optimal stiffener layout of an integrally stiffened panel. Tenek and Hagiwara (1994) applied the homogenization method in order to seek the optimal thickness distribution of isotropic and anisotropic plates. Among other example cases, Pedersen (2000) and Huang et al. (2010) applied the SIMP method and BESO heuristics, respectively, to fundamental natural frequency maximization of three-dimensional plates with both clamped and simply supported boundary conditions. However, they discretized the design domain into only a single layer of elements in the vertical direction (see the optimized design for simply supported boundary condition by Pedersen (2000) in Figure 2.14(a)). Huang et al. (2010), as well as Allaire and Jouve (2005), also defined a design problem with a three-dimensional design domain, which optimized result is shown in Figure 2.14(b).

Presumably, we could define a three-dimensional design domain under a fixed continuous panel structure and seek the optimal distribution of material using one of the above mentioned optimization methods. The approach would enable us to explore stiffener



(a) Plate structure, adopted from the paper by [Peder-
sen \(2000\)](#)

(b) Three-dimensional structure with a point mass at the center point of the bottom plane of the structure, adopted from the paper by [Huang et al. \(2010\)](#)

Figure 2.14: Example of designs with optimized fundamental natural frequencies obtained using the SIMP method (a) and BESO heuristics (b).

layouts in a significantly more flexible design space than those of iso- and orthogrids. However, the optimized design, obtained using this approach, would most likely contain complex geometric features. Although additive manufacturing methods are also established for metals, they are feasible only for relatively small-scale components of small production lots ([Frazier, 2014](#)). Other manufacturing methods for integrally stiffened (metal) panels, such as face milling and friction stir welding, are suitable for large-scale components with large production lots. However, the complex geometric features would most likely be beyond the constraints of these manufacturing techniques.

Evolutionary topology optimization methods offer an alternative approach where the manufacturability of the final designs (that of face milling or friction stir welding) may be ensured. [Inoue et al. \(2002\)](#) studied a stiffener layout problem of reducing the vibration and structure-borne noise of a gearbox housing. The authors first defined a ground structure of stiffeners attached to the outer surface of the housing, and then applied a GA to seek the optimal combination of the ground structure members.

Another method of seeking the optimal stiffener layout is referred to as the *bionic growth method* ([Ding and Yamazaki, 2004](#), [Li et al., 2014](#), [Ji et al., 2014](#)). The method mimics the growth process of branching systems in biology, such as those of trees and roots¹⁷. In the method, the stiffener layout is represented by a branching system. The optimization process involves iterative expansions and modifications in the topology and cross-sectional area distribution of the system, which are determined based on the contributions of its individual structural members to the design objective. The expansion process follows a network of possible paths, i.e. the ground structure. Although these methods, initiated from the ground structure, always yield optimized designs that are easy to manufacture (i.e. they consist of stiffener-like structural members), their design space is limited to a predefined set of candidate structural members.

¹⁷However, the method is not to be associated with those based on L-systems (Section 2.7).

L-systems-based encodings provide an alternative approach that enable of exploration of the stiffener layouts beyond a predefined ground structure, while still ensuring the stiffener-like geometry of structural members. Sabbatini et al. (2015) applied the L-systems and its turtle interpretation (Section 2.7.1) as a parameterization method to multi-objective stiffener layout optimization, where one of the objectives was to minimize the vibration amplitude of a panel structure. However, the map L-systems-based encoding, which is the other type of L-systems based encoding reviewed in this chapter (Section 2.7.3), could perhaps be more suitable for the purpose than the one based on the turtle interpretation. The argument is that the map L-systems-based encoding naturally yields genotypes consisting of structural members that are fully connected to each other. In other words, these genotypes do not contain ‘dead end’ stiffeners (which are typically produced by the turtle interpretation). In addition, the phenotypes from the map L-systems-based encoding can be conveniently mapped into the two-dimensional domain of the panel structure. As far as we are able to ascertain, the map L-systems-based encoding has not been applied to stiffener layout optimization.

2.9 Conclusions

The main purpose of this chapter was to review commonly-used topology optimization methods¹⁸ – with a special emphasis on their design space parameterization. In the literature, we identified a clear division of topology optimization methods into gradient-based and evolutionary (also non-gradient-based) methods, as well as a categorization of the parameterization methods into direct, geometric and generative encodings. While both gradient-based and evolutionary optimization methods have been applied to the first two parameterization methods, only evolutionary optimization methods have been applied to generative encodings (as far as we are able to ascertain).

Evolutionary topology optimization methods are described to have a significantly higher computational cost than gradient-based methods (Sigmund, 2011, Deaton and Grandhi, 2014, Munk et al., 2015). However, considering their benefits, listed in the introduction (Chapter 1), and the fact that they can easily be used in conjugation with generative encodings, our objective in this chapter was to identify possible ways to improve the performance of evolutionary topology optimization methods, which use either direct or generative encodings.

We found indications that multi-dimensional encodings could improve the performance of the ground structure approach (i.e. direct encoding), due to their better geographical linkage in comparison to traditional one-dimensional encoding (Section 2.3.1.1). Thus, we will investigate the use of two-dimensional encodings in Chapter 3.

¹⁸The field of topology optimization recognizes a multitude of different methods, from which we have reviewed those that we consider the most relevant for our applications.

L-systems-based parameterizations¹⁹ – especially those that are based on map L-systems – have gained popularity among topology optimization researchers. In Section 2.7.4, we found that the majority of studies with the map L-systems-based method used similar numerical representations that originate from that defined by [Pedro and Kobayashi \(2011\)](#). Considering that these evolutionary processes do not operate directly on the design, the optimal set of the evolutionary control parameters should be less dependent on the optimization problem than in direct encoding methods. However, no systematic efforts have been reported in the literature on understanding the effects of the control parameter choices on the performance of such optimization processes. Thus, we will perform a statistical experiment evaluating these effects in Chapter 4.

In the studies using the map L-systems-based method, the age of the system is typically parameterized as one of the additional variables. However, this approach is potentially disadvantageous to the evolvability of the method, as a small change in this variable may cause a major change in the corresponding phenotype. In Section 2.7.4, we proposed the approach of linearizing of the age variable, which we will further investigate in Chapter 5.

These conclusions are related to the first research aim of this work, i.e. to improve the performance of evolutionary topology optimization. To summarize these conclusions, we define the following hypotheses:

Hypothesis 1: The use of two-dimensional encoding improves the performance of the ground structure approach in comparison to one-dimensional encoding, because of its better geographical linkage between the genotype elements.

Hypothesis 2: The choice of (genetic) control parameters has a significant effect on the performance of the L-systems-based methods.

Hypothesis 3: The fact that the evolutionary process in L-systems-based method (contrary to direct encoding) does not directly operate on the design reduces the problem-dependency of its optimal control parameter combination.

Hypothesis 4: Linearizing the age variable in the map L-systems-based enhances the evolvability of the method and thus improves its performance.

We reviewed studies relevant to structural optimization of the aircraft wing structure in Section 2.8.1. We found that studies with the continuum material definition do not yield the typical spar-rib structures ([Stanford and Dunning, 2014](#)). In fact, the optimized designs often have three-dimensional features beyond the limits of conventional manufacturing techniques. The manufacturing of the optimized structure can be facilitated by using the ground structure approach or L-systems-based method and limiting

¹⁹In Section 1.1, we narrowed the scope of generative encodings to L-systems-based methods.

the design space to contain only sheet- and stiffener-type structural members. The map L-systems-based topology optimization studies on aircraft wings (Kobayashi et al., 2009, Kolonay and Kobayashi, 2010, 2015) consider only full-depth structural members; however, wings with the effective use of material typically consist of both full-depth and stiffener-like structural members.

Aircraft wings are prone to buckling, as they consist of slender structural members. However, the topology optimization methods, considering buckling as the objective or constraint function, have only been applied to simple small-scale optimization problems. In applications to the aircraft wing, very few studies have included the evaluation of the critical buckling load using an FE analysis of the whole wing structure. Often, if the critical buckling load is considered as a constraint, it is evaluated using analytical estimates for individual skin panels. The accurate evaluation of critical buckling loads already in the conceptual design is vital to prevent expensive design changes later in the preliminary or detailed design.

Next, we reviewed topology optimization studies on conductive heat transfer systems in Section 2.8.2. In the literature, the majority of the studies have been conducted on the two-dimensional benchmark problem defined by Bejan (1997). Nearly all of these studies yield optimized designs, which can be characterized as branching tree-structures, yet Dede (2009) report his designs to have self-similar features. L-systems and its turtle interpretation (Section 2.7.1) naturally yield phenotypes with these features. However, the method has not been directly applied to the optimization problem.

Finally, we reviewed studies relevant to the natural frequency maximization of integrally stiffened panels in Section 2.8.3. Traditionally, the stiffeners in these panels are arranged in the patterns of iso- and orthogrids, in order to ensure their convenient manufacturing and analysis. Topology optimization methods provide means of increasing the fundamental natural frequencies of these panels. However, those based on gradient-based methods yield designs with complex geometric shapes, impeding the manufacturing of the panels. On the other hand, the design space of the ground structure approach is limited to a predefined set of candidate structural members. Thus, we propose that the map L-systems-based method (Section 2.7.2) could be suitable for the purpose. As far as we are able to ascertain, the method has not been applied to the optimization problem.

The conclusions in the two previous paragraphs are related to the second research aim of this work, i.e. to find topology optimization applications, to which L-systems-based encodings are particularly suitable. We summarize them into the following hypotheses:

Hypothesis 5: The parameterization method based on L-systems and their turtle interpretation is particularly suitable to topology optimization of heat conductors, as it naturally yields bifurcating tree-structures with structural continuity.

Hypothesis 6: The map L-systems-based method is particularly suitable to the natural frequency maximization of an integrally stiffened panel, as it facilitates a convenient implementation of manufacturing constraints and naturally yields layouts with full stiffener continuity.

We will apply the ground structure approach and L-systems-based methods to the selected applications in Chapters [3-7](#).

Chapter 3

The ground structure approach to structural topology optimization

As mentioned in the introduction (see Sections 2.2.1 and 2.4), the ground structure is a finite, but large, subset of the structural universe, which is an infinite collection of all permissible structural members inside the design space. Ground structure members are candidates for the final optimized structure.

In this chapter, we use two alternative optimization methods to seek the optimal combination of the ground structure members. The first, following the main scope of the work, is a genetic algorithm (GA) (Section 3.4) and the second is a (bi-directional) evolutionary structural optimization ((B)ESO) heuristic (Section 3.5).

We place the ground structure approach into the context of structural design of the aircraft wing. First, we describe a generic framework of topology optimization of the aircraft wing internal structure using the ground structure approach, and then, in Section 3.7, apply the methods to a design of a small unmanned aerial vehicle (sUAV) wing internal structure, built via additive manufacturing.

3.1 Procedure

The objective of the optimization framework is to minimize the structural mass of the wing, m_{wing} , subject to von Mises stress and buckling constraints. Thus, the optimization problem is defined as

$$\begin{aligned} & \text{minimize} && m_{\text{wing}}(\mathbf{x}) \\ & \text{w.r.t} && x_i && i = 1, 2, \dots, n_d \\ & \text{subject to} && \sigma_{\text{vM},i}^{\max} \leq \sigma_{\text{vM}}^{\text{limit}} && i = 1, 2, \dots, n. \\ & && \bar{\lambda}_1 \geq 1, \end{aligned} \tag{3.1}$$

where $\mathbf{x}$ is a vector of n_d design variables, $\sigma_{vM,i}^{\max}$ is the maximum von Mises stress in *material section i* (with a total of n sections) and $\sigma_{vM}^{\text{limit}}$ is the maximum allowed von Mises stress. Here, we define a material section to be an individual ground structure member or a section of the skin bordered by the members of the ground structure. $\bar{\lambda}_1$ is the lowest normalized buckling load of the structure, defined as

$$\bar{\lambda}_1 = \lambda_1 / F_{\text{des}}, \quad (3.2)$$

where λ_1 is the lowest buckling load and F_{des} is the design load.

The optimization procedure is outlined in Figure 3.1. The inputs to the process are the geometric definitions of the outer mold line (OML) and the ground structure, as well as the loading of the wing. To start the process, the OML and ground structure generators are used to produce the OML geometry of the wing and a ground structure inside the OML, respectively. The ground structure involves a large number of structural candidates, from which the structural optimizer picks a set of candidates to the final, optimized design. The optimizers are, as already mentioned, a GA and a (B)ESO heuristic. During the optimization process, finite element (FE) analysis is used to evaluate the feasibility of individuals in terms of the stress and buckling constraints. The final output is the optimized structural topology and the corresponding structural mass.

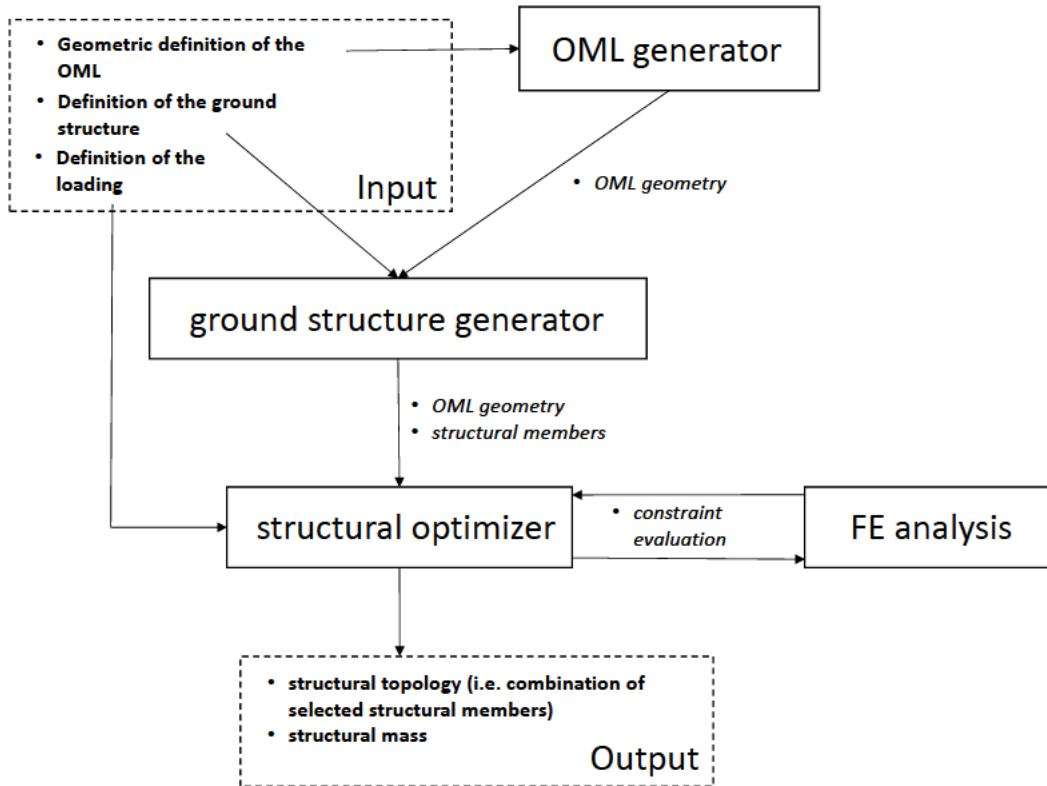


Figure 3.1: The topology optimization procedure via the ground structure approach.

3.2 Geometry generation

The geometries, in both the OML and ground structure generators, are generated automatically using parametric geometries. In our implementation, these geometries are NURBS (Non-Uniform Rational B-Splines) surfaces, generated in Rhinoceros using its Python scripting interface. The OML generator uses an open-source collection of Python objects, called AirCONICS (Aircraft Configuration through Integrated Cross-disciplinary Scripting)¹, developed by [Sóbestor and Forrester \(2014\)](#), [Sóbestor \(2015\)](#). The ground structure generator uses a newly developed set of Python objects. The modules are able to define the OML geometries and ground structures of both conventional and unconventional wings. Three visualized ground structures and corresponding OML geometries made with these modules are presented in Section 3.7.3.

3.3 Ground structures and the component hierarchy

The ground structure is a selection of structural members that are candidates for the final structure. Traditionally, these structural members all have the same type (e.g. a beam, shell or solid). We extend the ground structure to be a stencil, containing slots where structural members of different types can be assigned. The stencil is drawn in the two-dimensional planform shape of the wing. We use three types of stencils: quadrilateral, quadrilateral with diagonals and hexagonal (Figure 3.2). For the sake of brevity, we refer to the ground structure stencil simply as the ground structure. Further, we refer to a line connecting two nodes in the stencil (where ground structure members can be assigned) as a *ground structure slot*.

The inside of the aircraft wing, like most thin-walled structures, is often constructed using two types of structural components, which are a full-depth structural member extending between the opposite skins of the wing² and a stiffener attached to either of the skins. We use these to define a component hierarchy, from which structural members are drawn to the ground structure slots. In this work, we use the following component hierarchy, starting from the structural member providing the most support: (1) full-depth structural member with a lightening hole, (2) a stiffener on the upper skin and (3) no structural member (Figure 3.3). We have included a stiffener only on the upper skin in the hierarchy because the upper skin is more prone to buckling than the lower skin³.

The full-depth members are oriented perpendicular to the planform shape of the wing, and their lightening holes have an elliptical shape, which dimensions are determined

¹Available at www.aircraftgeometry.codes

²Traditionally, this type of structure is referred to as a spar or rib, depending on its orientation.

³The positive limit load factor is always greater (in absolute value) than the negative limit load factor.

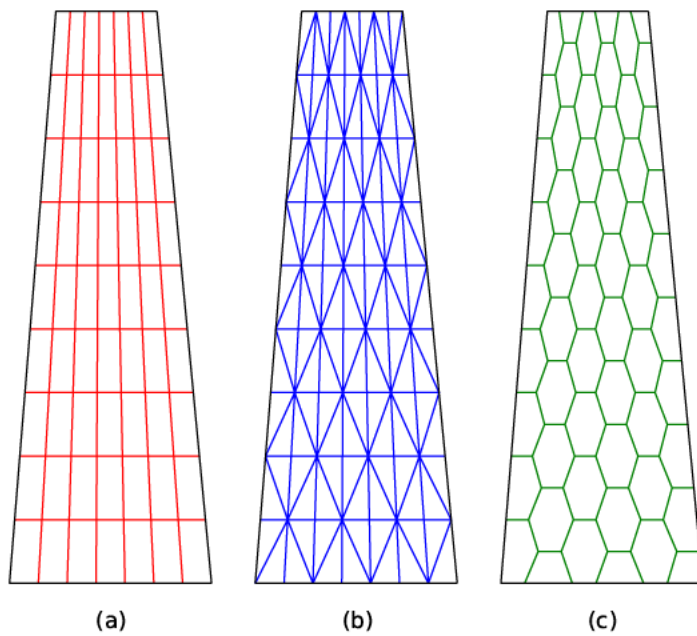


Figure 3.2: Quadrilateral (a), quadrilateral with diagonals (b) and hexagonal (c) ground structures.

as a fraction c_{lh} of the main dimensions of the member. The stiffeners are given a cross-section, which is aligned to be inside the wing, perpendicular to the skin panel.

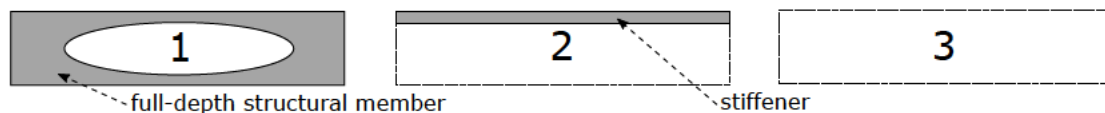


Figure 3.3: The component hierarchy.

The reasoning for the presented ground structures is the following. The quadrilateral ground structure (Figure 3.2(a)) is a superset of the components making up a traditional wing structure. If its ground structure slots are filled with full-depth structural members (index 1 in the component hierarchy), the resulting structure has features similar to spars and ribs, seen in conventional aircraft wing internal structures. The next ground structure (Figure 3.2(b)) introduces diagonal slots into the quadrilateral ground structure, to improve the flexibility of the parameterization. As mentioned in the literature review, some studies indicate that the addition of diagonal members may improve the efficiency of the final wing structure.

A final structure initiated from quadrilateral ground structures with or without the diagonal slots, have, in theory, a structural weakness. Let us consider a node in a ground structure that in the final structure connects three structural members, of which two are collinear and the third is perpendicular to the two collinear structural members, i.e. a T-junction. Structurally, the two collinear structural members provide very little

supporting stiffness to the perpendicular structural member. To avoid the formation of T-junctions, we align the slots in the third ground structure (Figure 3.2(c)) in hexagonal shapes. These shapes are fitted into the quadrilateral domain of the wing by introducing quadrilateral and pentagonal shapes at the domain boundaries.

Although here we specify the component hierarchy to contain only three options, a designer using the same approach could easily include more options in the hierarchy. Figure 3.4 presents an example of an extended component hierarchy, including also a (1) full-depth structural member without a lightening hole and (3) stiffeners on both skins.

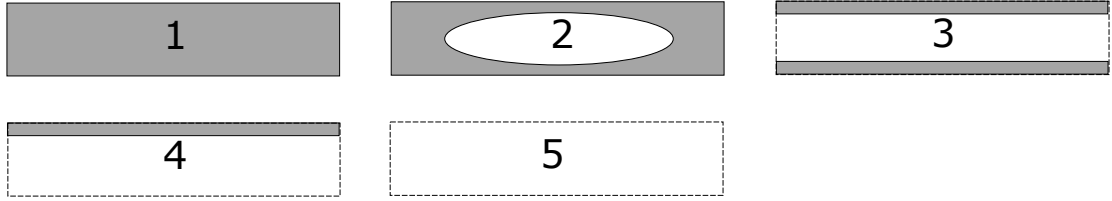


Figure 3.4: An extended component hierarchy.

3.4 GA-based optimization method

The current optimization problem has a multi-modal design landscape, because various subsets of the ground structure may have a similar objective function value. Genetic algorithms (GAs), reviewed in Section 2.3, are global search methods that have the potential of finding the optimal, or a nearly optimal, solution also in multi-modal design landscapes.

The ground structure approach, evolved using GAs, is criticized because it may produce designs where some structural members are not connected to the rest of the structure (Wang and Tai, 2005, Deaton and Grandhi, 2014, Munk et al., 2015). However, this is not an issue in our case as both full-depth and stiffener members are attached to at least one of the skins, i.e. the connectivity of a structural member is independent of the presence of other structural members.

As mentioned in the previous section, each ground structure slot can be assigned three different options, indexed from 1 to 3. We encode these indices directly to the elements of a genotype.

One of the objectives of this chapter is to study if using the two-dimensional encoding improves the performance of the GA-based topology optimization in comparison to the commonly-used one-dimensional encoding. Based on the positive indications in the literature (Section 2.3.1.1), the two-dimensional encoding is more likely to maintain the geographical linkage of elements close to each other in the ground structure. Figure 3.5

illustrates the geographical linkage in a crossover of two individuals, encoded on two-dimensional genotypes. The first parent has only the transverse slots of the quadrilateral ground structure filled with full-depth structural members, whereas in the second parent only the longitudinal slots are filled with either full-depth or stiffener members. As the number of spanwise structural member slots is one less than the number of chordwise member slots (in the chordwise direction), the last element of odd rows in the matrix is a dummy element, indicated by symbol 'X'. The crossover is conducted with a clear diagonal splice. As it can be seen, the offspring has a significant number of features from both different parents. While the crossover in Figure 3.5 is an artificial example designed to illustrate the principle behind the operator proposed here, the crossover operators in GAs are randomly defined.

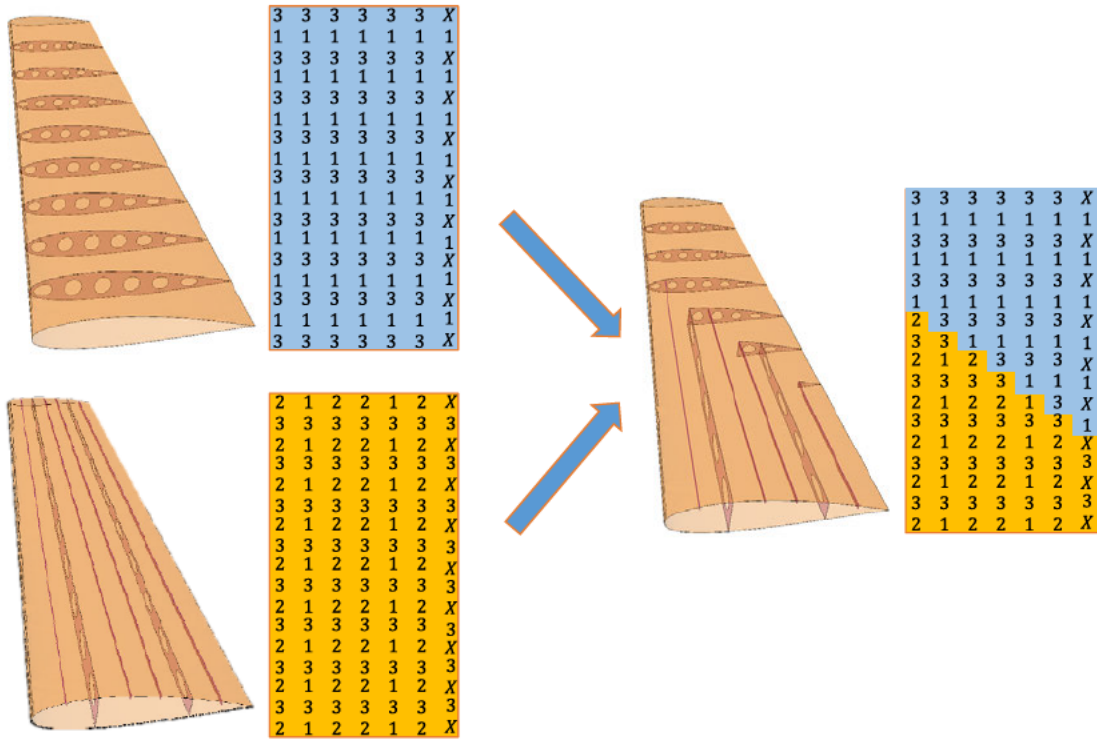


Figure 3.5: Example of crossover between two-dimensional strings representing wing designs.

We use two different crossover operators for one-dimensional genotypes, which are the two-point (Figure 3.6(a)) and uniform crossover (Figure 3.6(b)). These crossover operators were presented in Section 2.3.2.

The crossover of two-dimensional genotypes could be performed, in its simplest form, by slicing the genotypes either horizontally or vertically. However, this would lead to a low diversity of splicing strategies that a crossover operator can produce (Kahng and Moon, 1995). A better solution is to use more randomized operators. In this study, we define grid points in between the elements and create a randomized path through these points. The grid has a total of $(n_r + 1) \times (n_c + 1)$ points, where n_r and n_c are the dimensions of

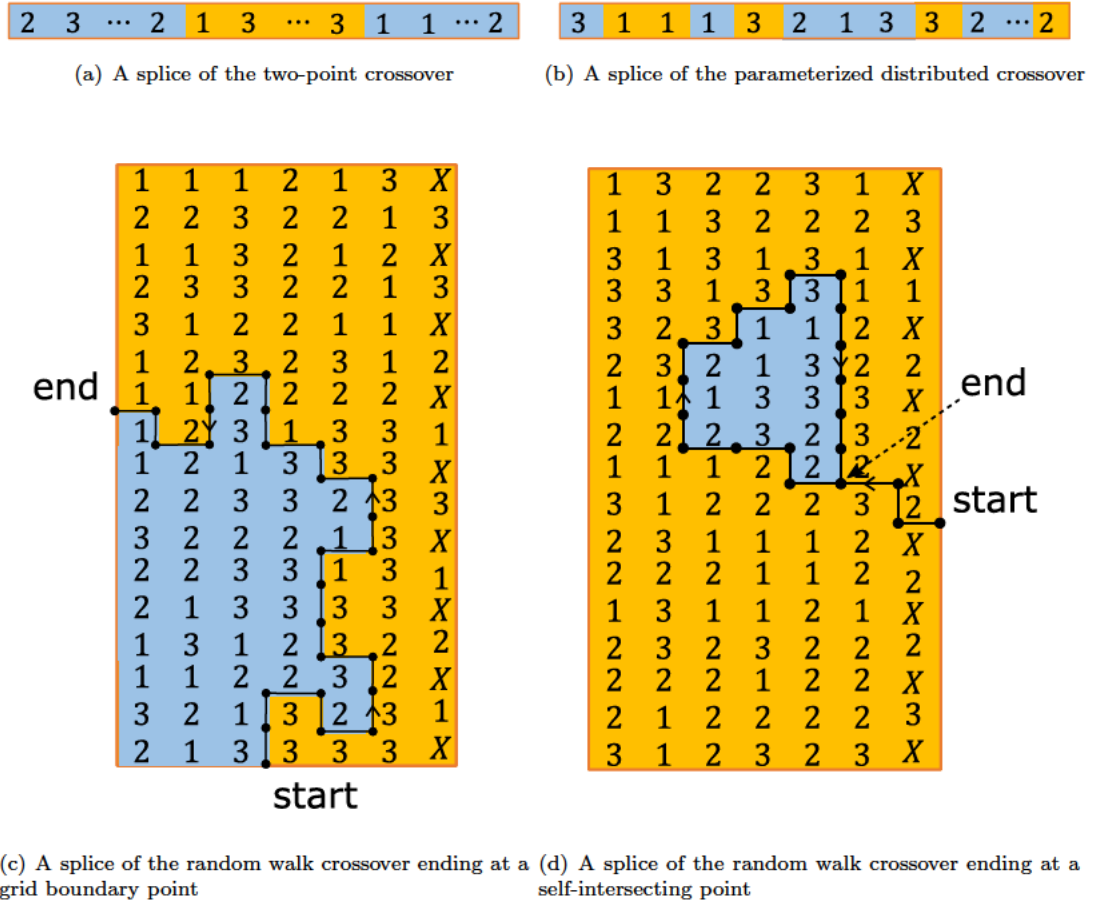


Figure 3.6: Example splices of crossovers on one- and two-dimensional strings, where X represents an element, and blue and yellow colors indicate genes inherited from parents 1 and 2, respectively.

the two-dimensional genotype. The random path is generated by first picking a random grid point at the domain boundary and then repeatedly moving to another grid point either left, straight or right with respect to the last two points. The process is terminated when another boundary is reached (Figure 3.6(c)) or the path becomes self-intersecting (Figure 3.6(d)). We shall term this operator *random walk crossover*.

We use two parameters to control the development of the path in the random walk crossover. First, p_s is the probability for the path to move to a point straight ahead of the current point. The remaining left and right turns have the same probability of $(1 - p_s)/2$. The threshold value $N^{\min}$ is used as a minimum fraction of elements from both parents in the offspring. If the threshold value is not reached, a new splicing path is generated. We tested several combinations of parameters p_s and $N^{\min}$, and observed that relatively diverse crossover operators can be obtained, while avoiding very small element proportions from a single parent, by using values: $p_s = 0.6$ and $N^{\min} = 0.05$.

The design problem is constrained by the allowable von Mises stress and critical buckling load. Earlier, in Section 2.3.3, we found indications in the literature that penalty

function methods are a suitable starting point when applying GAs to constrained optimization problems. In addition, a recommendation was found that repair algorithms are suitable for combinatorial optimization problems, to which our current optimization problem belongs. However, these algorithms require either a greedy search or a user-defined heuristic. The former would be computationally expensive with our FE analysis-based constraint evaluation, and we do not know a suitable procedure for the latter. Finally, recommendations were given for optimization problems with linear constraints and highly constrained design spaces. The constraints of the current optimization problem are not linear, and, because the number of constraints is relatively low, we assume the design space not to be highly constrained.

Due to these reasons, and for the sake of simplicity, we handle the constraints via the static penalty function method. Thus, we modify our objective function (Equation 3.1) to be

$$\begin{aligned} \text{minimize} \quad & m_{\text{wing}}(\mathbf{x}) + r_s \sum_{i=1}^n k_{s,i}(\mathbf{x}) + r_b k_b(\mathbf{x}) \\ \text{w.r.t} \quad & x_i \quad \quad \quad i = 1, 2, \dots, n_d, \end{aligned} \quad (3.3)$$

where r_s and r_b are penalty coefficients of violated von Mises stress and buckling constraints, respectively. The coefficients are adjusted so that the penalty terms have a similar amplitude as the objective term. Variables $k_{s,i}$ and k_b accommodate the possible violations of the stress and buckling constraints, respectively. The von Mises stress penalty is determined independently in the n sections of the structure. These variables are defined as

$$k_{s,i} = \begin{cases} \sigma_i^{\max} - \sigma^{\text{limit}}, & \text{if } \sigma_i^{\max} \geq \sigma^{\text{limit}} \\ 0, & \text{otherwise} \end{cases} \quad (3.4)$$

$$k_b = \begin{cases} 1 - \bar{\lambda}_1, & \text{if } \bar{\lambda}_1 \leq 1 \\ 0, & \text{otherwise.} \end{cases} \quad (3.5)$$

We have implemented the GA, described in this chapter, using Pyevolve (Perone, 2009), an open source library of evolutionary operators, capable of parallel processing and implementing a set of one- and two-dimensional crossover operators (to which we have added the random walk crossover).

3.5 (B)ESO-based optimization method

Before going into the details of ESO, let us define what the rejection of a member means in our current study. As described in Section 3.3, the ground structure slots are assigned a structural member from the component hierarchy, containing a total of three options. The order in the hierarchy reflects the extent to which an option strengthens the wing.

Therefore, we define the rejection of a structural member as moving one step down in the hierarchy (e.g. from index 1 to 2).

The determination of sensitivity numbers is critical to ensuring the effectiveness of ESO. In the literature, methods have been developed to determine sensitivity numbers for stress, displacement, buckling and frequency constraints. As mentioned earlier, the present study considers only von Mises stress and buckling constraints, so here we consider only these sensitivity numbers. For stress, the sensitivity number α_i of an element i can be simply defined as

$$\alpha_i = \sigma_{\text{vM},i}^{\max}, \quad (3.6)$$

where $\sigma_{\text{vM},i}^{\max}$ is the maximum von Mises stress of the element. The sensitivity number for buckling is normally defined as

$$\alpha_i = -\mathbf{u}_{i1}^T [\Delta \mathbf{k}_i] \mathbf{u}_{i1} \quad (3.7)$$

where $\mathbf{u}_{i1}$ is the eigenvector of the element i in the lowest buckling mode, and $[\Delta \mathbf{k}_i]$ is the change in the stiffness matrix of the same element (the derivation of the equation was presented in Section 2.5.1). However, if the sensitivity number is used in a topology optimization of a shell structure, the thickness distribution of the shell structure must be continuous (Munk et al., 2015). Therefore, individual elements cannot be rejected from the structure. This conflicts with the aims of the current study, wherein we consider wholesale changes in topology.

Because of this limitation, we choose an alternative approach for the evaluation of element sensitivities. At each iteration, the buckling sensitivities of the elements are determined by individually removing elements from the structure and comparing its lowest eigenvalue with the reference structure as

$$\alpha_i = \bar{\lambda}_i - \bar{\lambda}_{ref}, \quad (3.8)$$

where $\bar{\lambda}_i$ is the lowest normalized eigenvalue of the structure without element i , and $\bar{\lambda}_{ref}$ is the lowest normalized eigenvalue of the reference structure having all the remaining elements. It is to be noted that this approach is computationally more expensive than the classical way of determining the sensitivity numbers for all elements from a single FE analysis.

We also adopt an alternative definition of the rejection rate RR_j . The reason is that, if multiple members are rejected from a certain region of the ground structure in the same iteration, the critical buckling load may suddenly drop under the stopping criterion. Therefore, to have a better control of the number of rejected members, at each iteration N^{rej} members with the smallest sensitivities are rejected.

In ESO, the rejection of an element is irreversible, which may lead the optimization to a local optimum. To overcome this problem, Querin et al. (1998) introduced bi-directional

ESO (BESO), where rejected elements may be recovered back to the structure. Two formulations of BESO exist in the literature. The *soft-kill* formulation does not remove a rejected element but changes its elastic modulus to a small value, which reduces its effect on the stiffness matrix of the structure. The other formulation, called *hard-kill*, removes the entire element from the FE mesh. In this study, we explored the use of both ESO and BESO. With BESO, the hard-kill formulation is used in the FE analysis, and the maximum number of recovered elements per iteration is limited to N^{rec} .

3.6 Constraint evaluation (FE analysis)

The wing structure is required to withstand the applied loads without an occurrence of two classical failure mechanisms: yield and loss of structural stability. From an optimization point of view, these requirements are viewed as constraints. The failure mechanisms are measured as the maximum von Mises stress in a section and the critical buckling load of the structure. The purpose of the constraint evaluation is to check whether a design point lies in the feasible region, and, if not, to what extent are the constraints violated. The challenge with these constraints is that for a three-dimensional structure they are highly nonlinear and cannot be expressed analytically. Reasonable accuracy for the constraint evaluation is achieved with finite element (FE) analysis, which is a well-established method for the numerical analysis of structural mechanics. The von Mises stress distribution is determined by a static analysis, and the critical buckling load using eigenvalue analysis.

FE analyses are performed in Abaqus using its Python-based scripting interface. The pre- and post-processing of the FE analysis are fully automated, because a large number of constraint evaluations are required during the optimization process. The pre-processing script generates an FE mesh of the geometries that were produced by OML and ground structure generators, assigns materials to its sections and sets the loads. The structural members that are included in the FE model are determined by the structural optimizer. The post-processing script fetches the von Mises stress distribution and the lowest critical buckling load, and returns them to the structural optimizer.

The full-depth structural members, as well as the skin sections, are modeled as shell elements, whereas the stiffeners are modeled as beam elements. The skin sections are meshed using a quad-dominated algorithm (produces both triangular and quadrilateral elements). When applicable, the skin sections are meshed with quadrilateral elements only, which are computationally more efficient. The full-depth structural members are meshed with triangular elements. The reason is the opening hole in the middle of the member, which makes quad or quad-dominated meshing infeasible. The beam elements are located at the root of the stiffener, so that their nodes are aligned with the corresponding nodes of the shell elements. The translational and rotational displacements of

each of resulting node pairs are constrained to be equal. The beam elements are assigned a cross-section with an L-shape. All elements are first order elements, so triangular and quadrilateral elements have three and four nodes, respectively.

It is well known that the mesh density is a trade-off between accuracy and computational cost; accurate results are obtained only with a high computational cost. Therefore, we will determine a sufficient mesh density for our application later in Section 3.7.4.1.

The wing is attached to the fuselage of the aircraft from its root. However, in order to reduce the computational cost, we exclude the fuselage from our FE model. Thus, we define a simply supported boundary condition for the nodes lying in the root plane of the wing (including also those that belong to the internal structure). We note that the decision of using the simply supported boundary condition is conservative as, in reality, the fuselage provides bending stiffness for the root elements.

Two types of loads are applied to the FE model: an inertial load f_g as a body force on all elements of the model and an air load as a pressure load f_p on the upper and lower skins. The loads are visualized in Figure 3.7. The inertial load acts on every element of the model, and has magnitude $m_e n_L g$, where m_e is the mass of the element, n_L is the limit load factor and g is the gravitational acceleration.

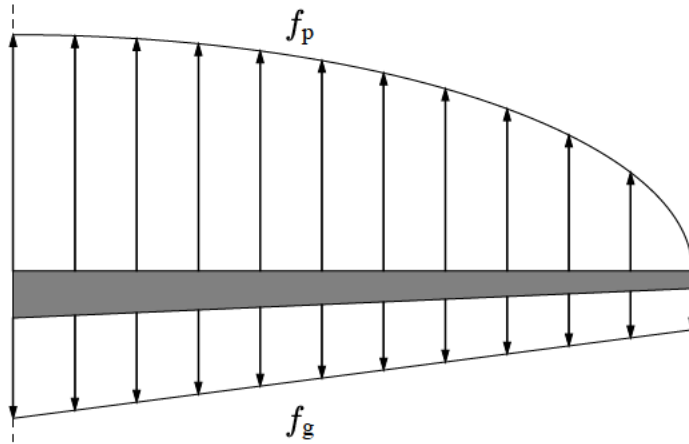


Figure 3.7: The pressure load f_p and inertia load f_g applied to the wing.

The air load has an elliptical load distribution along the wing span. The total magnitude of the air load corresponds to the take-off mass m_{TO} subtracted by the wing mass under the limit load factor n_L . We make the following two simplifications to the load distribution. First, we simplify the total load to be split evenly on the upper and lower surface of the wing. Second, we discretize the elliptic distribution in the spanwise direction into segments, and simplify the pressure distribution, in each segment, to be constant in the chordwise direction.

3.7 Application I: Topology optimization of a sUAV wing

In this section, we apply the ground structure based topology optimization methods to the design of a small Unmanned Aerial Vehicle (sUAV) wing, 3D printed in nylon. The task is to design the internal structure, inside the OML, that minimizes the structural mass of the wing subject to buckling, stress, and manufacturing constraints. The final results are benchmarked against corresponding results obtained by traditional design methods.

3.7.1 Description of the design problems

The choice of 3D printing as the manufacturing technique sets two main geometric constraints for the design: maximum bounding box dimensions and minimum wall thickness. In the following, we will refer to these as *manufacturing constraints*.

In this study, we choose the material of the wing to be nylon, which is commonly used in additive manufacturing. Its properties are listed in Table 3.1. Nylon, as well as other 3D printed materials, have anisotropic material properties depending on the direction in which they are layered. [Majewski and Hopkinson \(2011\)](#) studied the material properties of laser sintered Nylon-12 rods using tensile tests, in which the thickness and layering orientation of the rods were varied. They described the tensile properties of the material to be robust to changes in material thickness and build orientation. Moreover, they estimated that Young’s modulus and tensile strength in the weakest direction were roughly 80% and 94%, respectively, of the corresponding properties in the strongest direction. Therefore, and for the sake of simplicity, we assume the material to have homogeneous mechanical properties.

Parameter	Value	Unit
Young’s modulus E	1700	MPa
Poisson’s ratio ν	0.39	-
yield strength σ_y	48	MPa
density ρ	930	kg/m ³
min wall thickness $t^{\min}$	0.7	mm
max bounding box	650 x 350 x 550	mm

Table 3.1: Material and manufacturing properties of 3D printed nylon⁴.

Ensuring that each of the wings can be printed as a whole, the aircraft is chosen to have a semispan of 650 mm, which is equal to the maximum bounding box edge length. The geometric definition of the wing is given in Table 3.2. The wing profile is defined using two 4-digit NACA profiles, transitioning linearly from the root to tip.

⁴www.shapeways.com (accessed on 10th March 2016)

Parameter	Value	Unit
semispan $b/2$	650	mm
aspect ratio $\mathcal{R}$	7	-
total wing area S	0.241	m ²
taper ratio λ_t	0.5	-
sweep angle at leading edge Λ_{LE}	5.0	deg
dihedral angle Γ_{wing}	0.5	deg
root profile	NACA2420	-
tip profile	NACA2412	-

Table 3.2: Geometric definition of the sUAV wing.

The wing is required to withstand the loads without buckling or yielding under positive ($n_L = 4.5$) and negative ($n_L = -1.0$) limit load factors (Table 3.3). To evaluate these constraints, the following loads are applied to the FE model of the wing. First, a pressure load is applied to the upper and lower surfaces of the wing to describe the aerodynamic forces. The pressure corresponds to a maximum take off weight m_{TO} of 2.41 kg assuming that the weight of the wings, which is excluded from the pressure load, is 15% of the maximum take-off weight. The maximum take-off weight was chosen based on the total wing area S using a wing loading of 10 kg/m², which is realistic for a sUAV. The pressure load has an elliptical load distribution along the wing span. Second, inertial loads are applied on the wing weight as body forces with an amplitude $n_L g$, where g is the gravitational acceleration. Under a positive load factor, the inertial loads act in the opposite direction to the pressure load, and therefore decrease the total loading on the wing.

Parameter	Value	Unit
max take-off weight m_{TO}	2.41	kg
load factor range n_L	-1.0...4.5	-
wing loading m_{TO}/S	10	kg/m ²
factor of safety c_{FoS}	1.5	-

Table 3.3: Load properties of the sUAV.

As the first step of the design task, the *lightest manufacturable design* was analyzed, in which no internal structure was placed inside the OML and the entire skin was defined to have a thickness equal to the minimum wall thickness. This design is clearly the lightest design that still satisfies the manufacturing constraints. With the positive limit load, the maximum von Mises stress of the design is 7.48 MPa, which is, by a good margin, less than the yield stress of the material (Table 3.1). However, the structure buckles when only 44.0% of the design load is applied (Figure 3.8(a)). With the negative limit load, the maximum von Mises stress is 1.66 MPa, and the structure buckles at 93.0% of the design load. The shape of the buckling mode has its greatest values in either the upper or the lower skin near the root depending on the load factor (the positive limit

load causes buckling of the upper skin). In both cases, the buckling occurs first near the trailing edge where the curvature of the skin is the smallest (see Figure 3.8(a) for the buckling mode under the positive limit load).

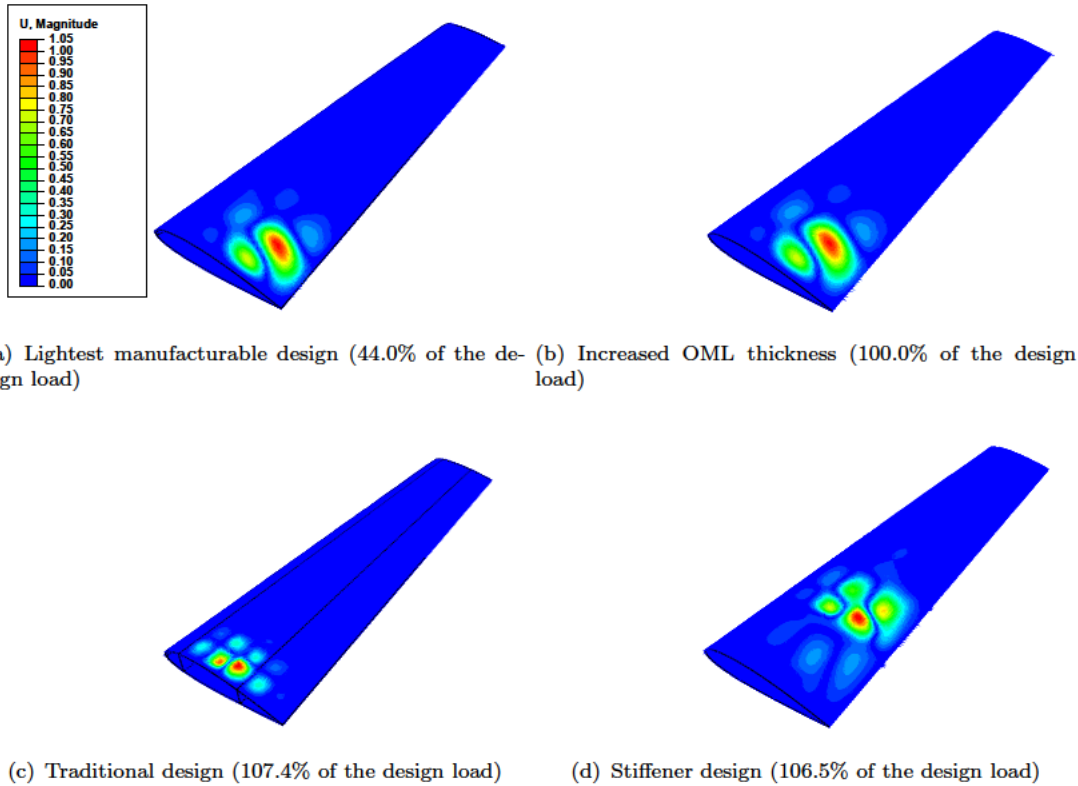


Figure 3.8: Modal shapes of the first critical buckling load for wing designs with traditional methods.

As a conclusion of the analysis of the lightest manufacturable design, the structure is not likely to yield under positive or negative limit loads with any internal structure arrangement. Thus, we omit the evaluation of the stress constraint in the optimization process. In contrast, the buckling constraint under the positive limit load is clearly an active constraint for the optimization and needs to be evaluated in the optimization process. Under the negative limit load, the skin alone with only the minimum wall thickness is almost strong enough to resist buckling. When the structure is stiffened to resist buckling under the positive limit load, the critical buckling load under the negative limit load is likely to be above the design load. Thus, we also omit the buckling constraint evaluation under the negative limit load from the optimization process, to reduce the computational cost of the analysis. To ensure the feasibility of the final design, all excluded constraints are verified after the optimization process.

3.7.2 Conventional design methods

The buckling resistance of a wing structure can be improved by increasing its skin thickness or by adding structural members, such as spars, ribs and stiffeners, inside the wing. This section presents three conventional design methods, starting from the simplest, to meet the required buckling strength. The results are benchmarks for the ground structure approach we use in this chapter.

The simplest way to provide the required buckling strength for the sUAV wing is to increase its skin thickness, which, in our case, has been set to the lower manufacturing constraint in the lightest manufacturable design. This approach is simple but obviously will not yield the best structure. Using a skin thickness of 0.956 mm, the normalized critical buckling load under load factor $n_L = 4.5$ becomes unity and therefore the design is feasible. Since only the skin thickness was varied, the modal shape of the critical buckling mode (Figure 3.8(b)) is almost identical to the lightest manufacturable design. The weight increase with respect to the lightest manufacturable design is 41.4 g.

Increasing the skin thickness is a rather naive way to fulfill the design criteria. A better solution is to stiffen the structure with spars, ribs and stiffeners. In the next design approach, which we refer to as the *traditional design*, two spars are placed at 15% and 65% of the chord, respectively, and four ribs are evenly distributed in the spanwise direction (Figure 3.9(a)). Finally, several FE iterations were performed to find the minimum number of spanwise stiffeners that provide a feasible design. In the critical buckling mode (Figure 3.8(c)), the greatest displacements are near the root of the upper skin, which buckles between the spars and stiffeners. In comparison to the lightest manufacturable design, the structural weight is increased by 32.7 g. Let us refer to this added mass as the internal structure mass m_{IS} .

Due to the manufacturing constraint, the skin thickness in this example is relatively high. Thus, the traditional design involving spars and ribs may be over-sized for the purpose. Therefore, in our last manual design approach, which we refer to as the *stiffener design*, we attempt to use only spanwise and chordwise stiffeners. After several iterations, a design was obtained where six spanwise stiffeners and two rims of chordwise stiffeners are located near the root of the wing (Figure 3.9(b)). Each of the chordwise stiffener rims extends around the whole wing profile. The lowest critical buckling mode of the design is plotted in Figure 3.8(d). Since the weight increment compared to the lightest manufacturable design is only $m_{IS} = 7.67$ g, the stiffener design is significantly better than the increased skin thickness design or the traditional design.

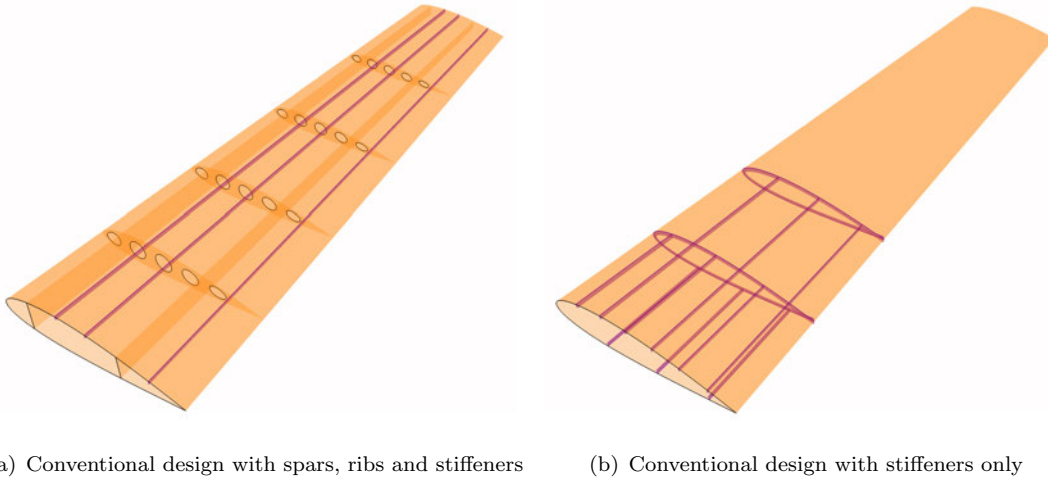


Figure 3.9: Two examples of conventional internal structures. The internal structures have been designed manually using conventional spars, ribs and stiffeners.

3.7.3 Ground structures for the application

We initiate the optimization processes from the three types of ground structures, described in Section 3.3.

The first (GS1) is a simple quadrilateral ground structure with six longitudinal and eight transverse sets of ground structure slots. Figure 3.10(a) presents an example structure obtained by assigning a full-depth structural member into all ground structure slots. The stiffeners, if assigned to a slot, have an L-profile of 4 x 0.8 mm, and a thickness of 0.7 mm, which is the same as the minimum wall thickness. The lightening hole in a full-depth structural member has a size fraction of $c_{lh} = 0.6$ (see Section 3.3). The total number of member slots in GS1 is 110.

GS2 (Figure 3.10(b)) incorporates two improvements over GS1. First, the diagonal slots are included in the ground structure, while keeping the number of longitudinal and transverse sets of members the same as in GS1. Second, the transverse slots are placed in a geometric series, where each transverse slot gap is 1.1 times the previous one (starting from the root). This modification shifts ground structure slots towards the root of the wing, where more internal structure is typically needed. The total number of member slots in GS2 is 173.

GS3 (Figure 3.10(c)) is a hexagonal ground structure, where the formation of potentially weak T-junctions is prevented (see Section 3.3). With this ground structure type, we also examine how its density affects the final, optimized objective function value. As mentioned earlier, a ground structure is a finite subset of the infinite structural universe, containing all possible structural members inside the design domain. The more member slots are included in a ground structure, the wider design space is explored and, therefore,

the better final designs are expected. We test GS3 with four different ground structure slot densities, which all have the same qualitative topology (Figure 3.11).

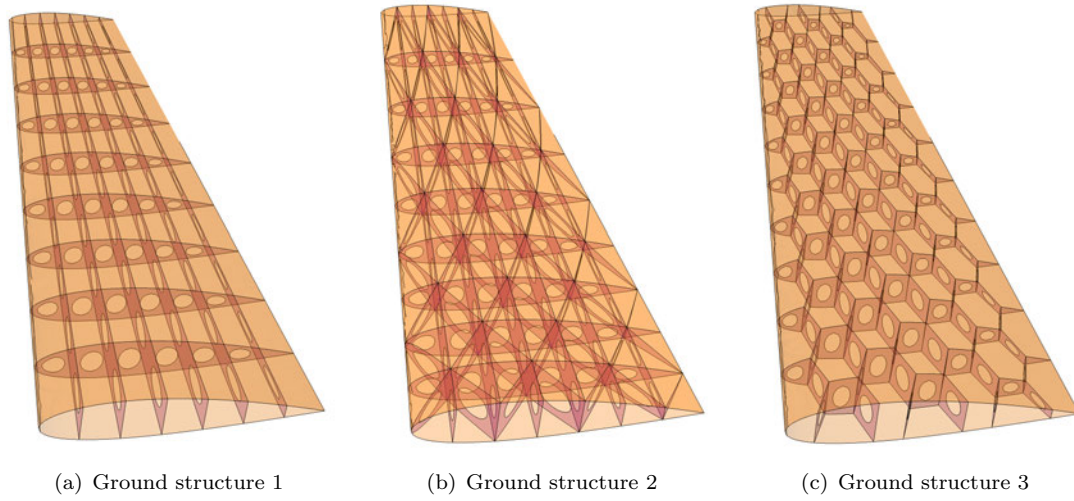


Figure 3.10: Three ground structures used in the study. The member density of Ground structure 3 is varied (see Figure 3.11).

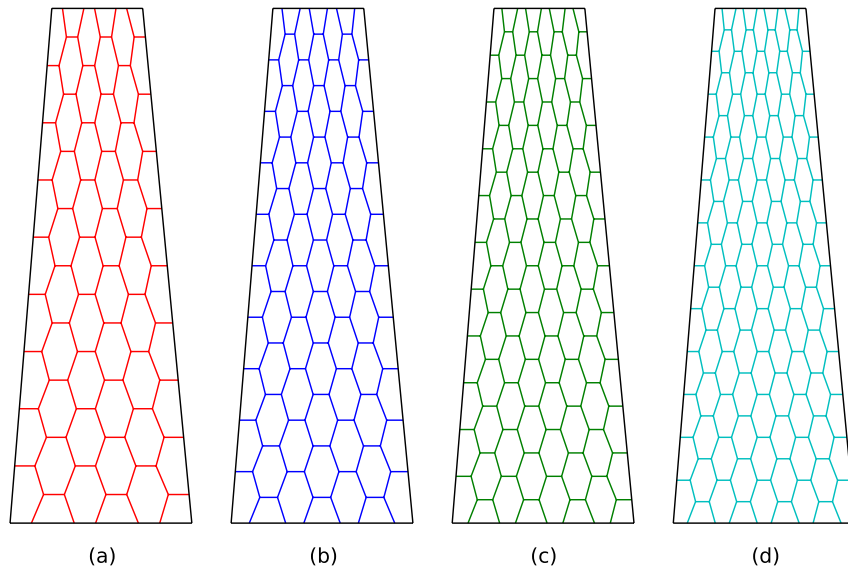


Figure 3.11: Four member densities for ground structure 3. The ground structures have a total of 141 (Subfigure a), 186 (b), 238 (c) and 295 members (d).

3.7.4 Results and discussion

The aforementioned topology optimization methods are deployed on the three ground structures, GS1, GS2 and GS3, defined in Section 3.7.3, from which GS3 has four alternative ground structure densities. The content of this section is as follows. First, in Section 3.7.4.1, we examine the dependency of the critical buckling load to the mesh

density of the FE model. Second, we use GS1 to adjust essential parameters of both GA- and (B)ESO-based optimization methods. Results of these parameter studies are presented for the GA-based method in Sections 3.7.4.3 and 3.7.4.4 and for the (B)ESO-based optimization in Section 3.7.4.5. Third, optimization runs with GS2 and GS3 are performed by exploiting the gathered parameter information from GS1. Finally, the optimization methods, and the ground structures, are compared to each other and benchmarked against the conventional design methods in Section 3.7.4.9.

3.7.4.1 Mesh density verification

The results of an FE analysis are dependent on the mesh density of the FE model. An inadequate mesh density stiffens the structure and, thus, the modeled critical buckling loads are overestimated. On the other hand, unnecessarily dense mesh increases the computational cost of the analysis. To assure effective evolutionary optimization, which involves a large number of objective function evaluations, it is essential to find a mesh density that provides sufficient accuracy with as low a computational cost as possible.

We have studied the critical buckling load with five mesh densities. The mesh densities are applied to FE models of both the full (all member slots filled with full-depth structural members) GS1 and an optimized wing structure initiated from the GS1. These FE models represent the two extremes of the feasible region of the design space. The obtained critical buckling loads are presented as a function of a representative mesh size in Figure 3.12. The critical buckling load of the full GS1 converges to a greater value than the optimized wing structure, due to the significantly greater number of structural members. In addition, the full GS1 requires a finer mesh than the optimized structure. The reason is that the mode of the critical buckling load of the full GS1 is more localized than the corresponding mode of the optimized wing structure. When choosing the representative mesh size, it is to be noted that inaccuracy in the constraint function values at the initial stages are more acceptable than at the final stages.

Here, we choose the representative mesh size of 5 mm for the remaining of the chapter. With this value, the critical buckling load is estimated to be 7.21% and 2.21% greater than with the smallest studied mesh size (1.25 mm) for the full GS1 and optimized wing structure, respectively. The computational cost with the chosen mesh size is significantly smaller than with the smallest studied mesh size.

Next, we investigate the effect of the penalty function on the evolutionary process of a single GA-based optimization run and conduct sensitivity studies with respect to two genetic parameters: the crossover type and the population size.

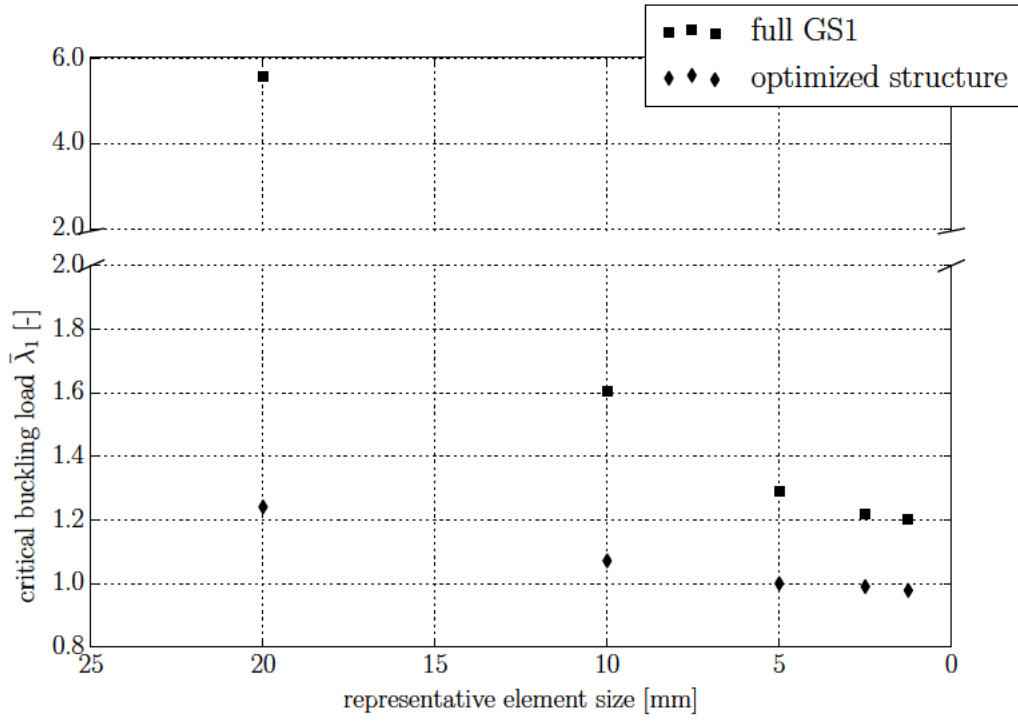
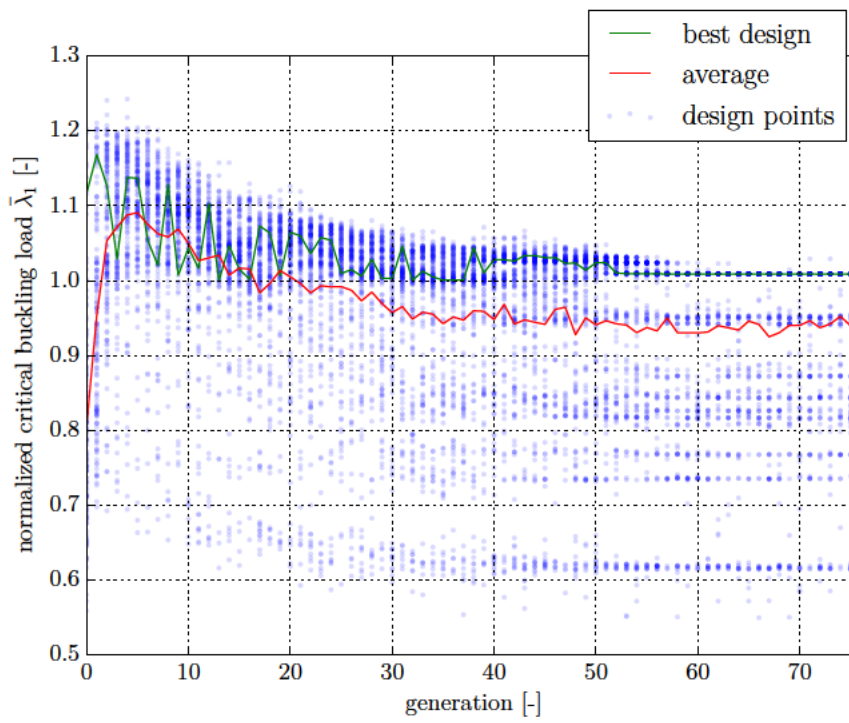


Figure 3.12: Mesh density verification of the full GS1 and an optimized ground structure.

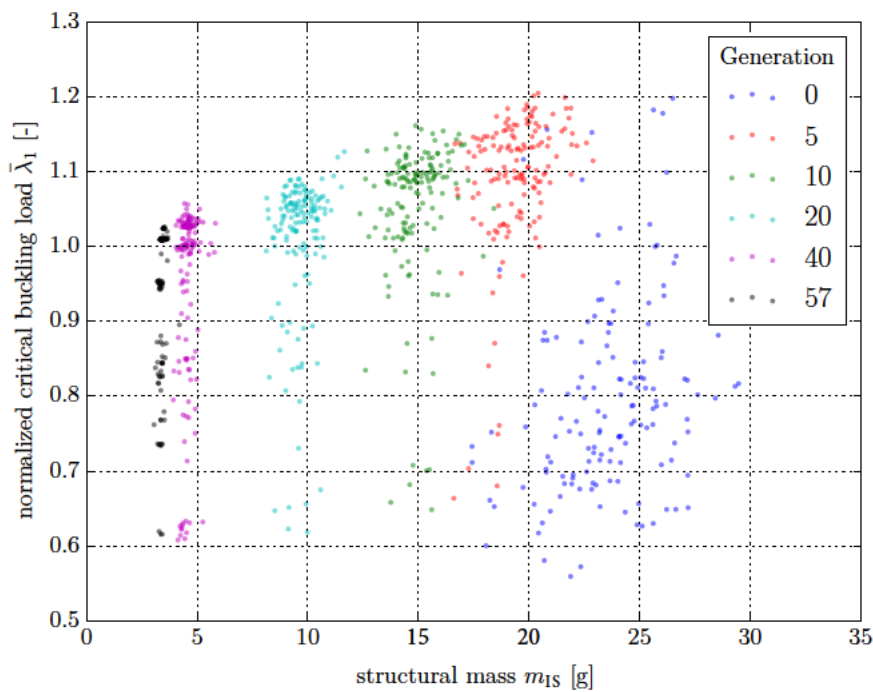
3.7.4.2 The use of penalty function in GA-based optimization

A penalty is applied to the objective function values of individuals violating the buckling constraint. The stress constraint is not evaluated (see reasoning in Section 3.7.1). We tested the optimization method with penalty coefficients $r_b = \{0.1, 1, 10\}$. The aim was to find a value for the penalty coefficient so that the objective and penalty terms in Equation 3.3 have roughly the same amplitude. Out of the three tested values, $r_b = 1$ was found to be the best for the purpose.

To illustrate the effect of the penalty function, let us examine a single optimization run on the GS1 using two-point crossover and a population size of 150. Figure 3.13 presents the constraint function values as a function of both the individual's generation (a) and the objective function value (b). Further, Figure 3.14 visualizes the evolution of the entire population as a contour plot, where infeasible individuals are marked with grey. As we can see, in the zeroth generation, in which all individuals are randomly generated, most of the individuals are infeasible. However, during the next generations the penalty function biases the search towards feasible designs; this is seen as a rapid increase in the number of feasible individuals per generation. The number of feasible individuals reaches its maximum at the fourth generation, after which it starts to decrease. We believe this behavior to be caused by the selective pressure driving the individuals towards ground structure subsets with smaller numbers of members. When fewer members are included



(a)



(b)

Figure 3.13: Representation of individuals in a GA-based optimization run. Subfigure (a) presents the normalized critical buckling load of individuals as a function of their generation, while subfigure (b) plots the individuals of selected generations based on their critical buckling load and structural mass.

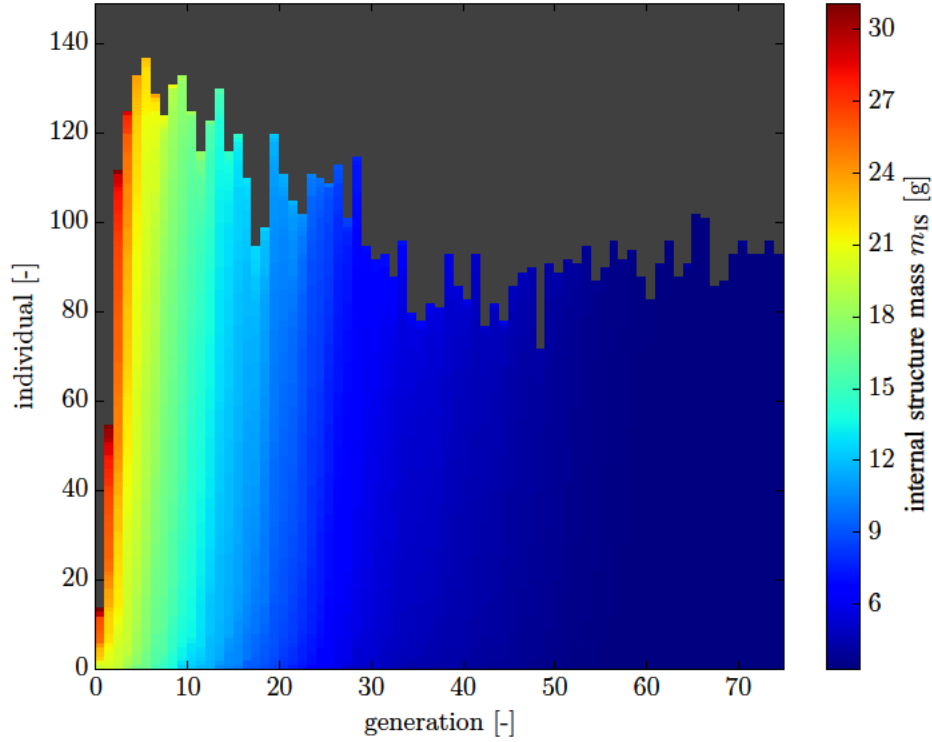


Figure 3.14: Contour plot of the score of all feasible individuals during the optimization process. Infeasible individuals are indicated by grey.

in the individuals of a generation, the removal of a critical member by the genetic operators (and therefore the creation of an infeasible individual) becomes more likely.

3.7.4.3 Population size in the GA-based method

The determination of a reasonable population size N_{pop} is critical in GA-based optimization, especially if the objective or constraint function evaluation is computationally expensive. Too small a population may cause a loss of diversity among the individuals, which may lead to a premature convergence to a local optimum. On the other hand, an over-sized population may not converge within the time constraint of the optimization process.

We run experiments here with population sizes $N_{\text{pop}} = \{38, 75, 150, 300\}$, while keeping other control parameters of the GA constant. The crossover and mutation rates are 0.9 and 0.02, respectively, and no elitism is used. Further, the tournament selection with a pool size of four is used, and the mutation operator is set to swap the locations of two randomly selected elements of the string.

The convergence histories of the optimization runs are presented in Figure 3.15, and the corresponding numerical data in Table 3.4. As we can see, optimization runs with

a population size of 300 provided the lowest internal structure masses. When the population size is decreased to 75 or less, some optimization runs converge prematurely to a local minimum. However, more optimization runs would be needed for statistically significant conclusions on the adequate population size. The two-point crossover is used in all of these optimization runs.

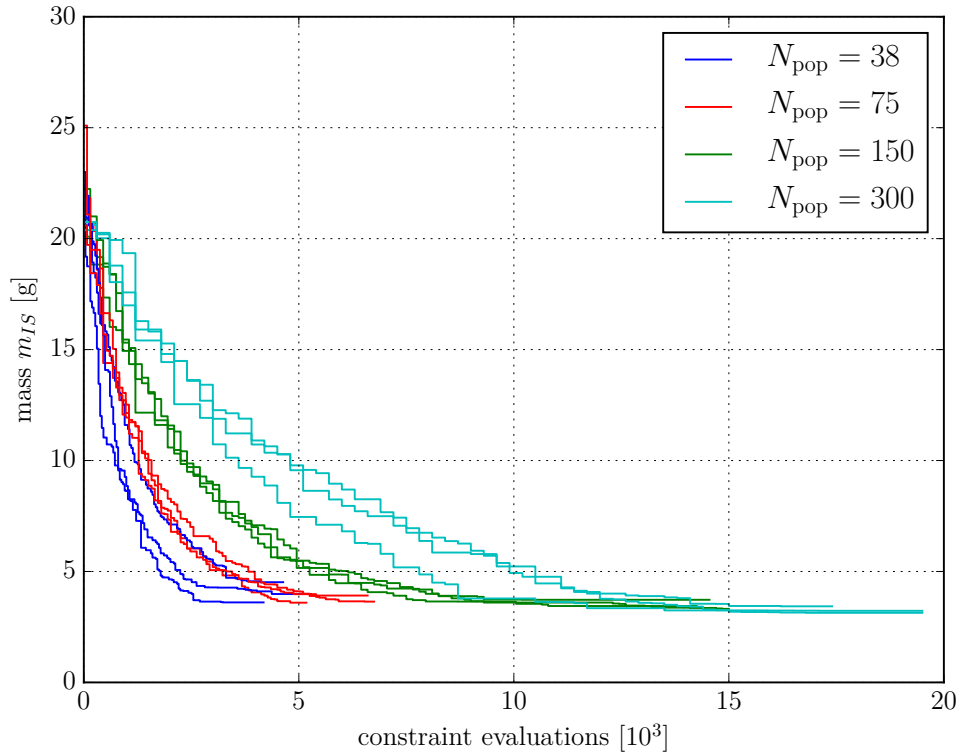


Figure 3.15: Convergence histories of experiments with different population sizes N_{pop} .

		population size N_{pop}			
		38	75	150	300
mass m_{IS} [g]	run1	3.985	3.593	3.443	3.430
	run2	4.516	3.915	3.314	3.228
	run3	3.606	3.642	3.724	3.136
	average	4.036	3.717	3.494	3.265
constraint evaluations [10^3]	run1	5.28	5.18	14.25	17.40
	run2	4.64	6.60	15.00	19.50
	run3	4.18	6.75	14.55	19.50
	average	4.70	6.18	14.60	18.80

Table 3.4: Numerical data of the experiment shown in Figure 3.15.

3.7.4.4 Crossover types in the GA-based method

Next, we present results with three alternative crossover types, which are two-point crossover in one-dimensional encoding, the random walk crossover in two-dimensional encoding and the distributed crossover (Section 2.3.2). Based on the result of the previous section, we choose to use a population size of 150. Other parameters of the GA are kept constant.

Since GAs are stochastic, multiple optimization runs are required with each crossover operator to obtain statistical evidence on whether one crossover operator has a better performance than another. The computational cost of a single optimization run is roughly 48 hours on a standard desktop (with a 4-core Intel® Xeon® W3520 processor). Thus, we choose to run 10 experiments with each of the crossover operator. Finally, let us assume that an optimization is converged when no improvement in the objective function value is obtained during 15 consecutive generations. We study two quantities of the obtained convergences: the number of generations required for a convergence, n_{con} , and the optimized mass m_{IS} .

From the three data sets with different crossovers, we define a family of six statistical tests, which are the two aforementioned quantities tested on the three pairwise combinations of the crossovers. The null hypothesis H_0 in each test is that the mean values of a quantity are similar ($\mu_1 = \mu_2$) with the two compared crossovers, whereas the alternative hypothesis H_1 is that the mean values are dissimilar ($\mu_1 \neq \mu_2$). We use the significance level of $\alpha_s = 0.05$.

When conducting a family of statistical tests, the probability of Type I error⁵ is inflated (Arcuri and Briand, 2014). A classical method of treating the inflated probability of Type 1 error is to use the Bonferroni correction, in which the significance level for each individual test in the family is defined as α_s/n , where n is the number of tests. However, the Bonferroni correction is considered overly conservative (Perneger, 1998, Nakagawa, 2004, Arcuri and Briand, 2014), and, therefore, we treat the inflated probability of Type I error by examining the step-up false discovery rate (FDR) (Benjamini and Hochberg, 1995), which is a less conservative variation of the Bonferroni correction. The step-up FDR is defined as: the expected proportion of true null hypotheses among all rejected null hypotheses is equivalent to the significance level α_s (Benjamini and Hochberg, 1995). The procedure of determining the step-up FDR is the following. First, the p-values of the statistical tests are ranked in decreasing order of significance, j being the resulting rank. Second, the p-values are examined in the opposite order, i.e. $j = n \dots 1$, and when finding the first p-value, ranked as the k th test, that satisfies

$$p_j \leq \frac{j\alpha_s}{n}, \quad (3.9)$$

⁵The type I error refers to the rejection of the null hypothesis when it is actually true.

the process is terminated. All null hypotheses ranked $j \leq k$ are rejected, and hypotheses ranked $j > k$ are accepted.

The convergence histories of the optimization runs are plotted in Figure 3.16. Means and sample standard deviations (SD) of the two quantities on different crossover types are listed in Table 3.5. The statistical data, along with corresponding normal distributions, are presented in Figure 3.17. However, considering the outliers in the statistical data and the relatively small sample size ($n = 10$), the normal distribution can only be used as an approximation of the probability distribution of a quantity. We do not know if the data is, in fact, normally distributed. Therefore, we test the statistical evidence using the non-parametric Mann-Whitney U-test.

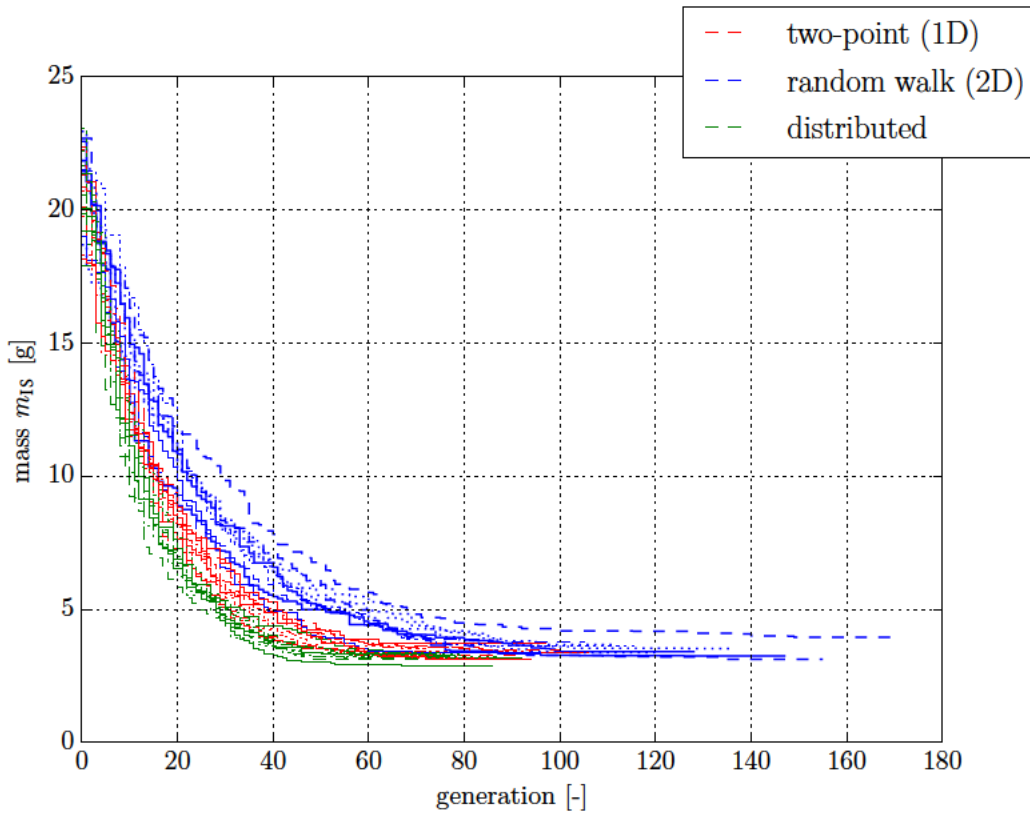


Figure 3.16: Comparison of the two-point (one-dimensional encoding), random walk (two-dimensional encoding) and distributed crossover.

Crossover	Generations to convergence n_{con}		Optimized mass m_{IS}	
	Mean	SD	Mean	SD
Two-point (one-dimensional)	83.70	7.40	3.38	0.186
Random walk (two-dimensional)	121.50	23.98	3.51	0.253
Distributed	74.80	7.83	3.21	0.141

Table 3.5: Mean and sample standard deviations (SD) of the required generations to convergence and the optimized mass with the tree tested crossovers. Each of the algorithms using these crossovers is tested 10 times.

The results of the six statistical tests are ranked, in decreasing order of significance, in Table 3.6. In each test, the crossover pair is ordered so that the crossover having, on average, the better performance is listed as crossover 1, and the crossover with a worse performance as crossover 2. Further, U is the U-value of the Mann-Whitney U-test, and p is the corresponding significance. The last column is the corrected significance level (Equation 3.9). Proceeding in the order $j = 6, 5, \dots, 1$, the first test satisfying Equation 3.9 is the test ranked 4th ($k = 4$). Therefore, we reject the null hypothesis H_0 in tests $j = 1, 2, \dots, 4$, and accept the alternative hypothesis H_1 meaning that the means of the quantities are dissimilar. In tests $j = 5, 6$, we fail to reject the null hypothesis.

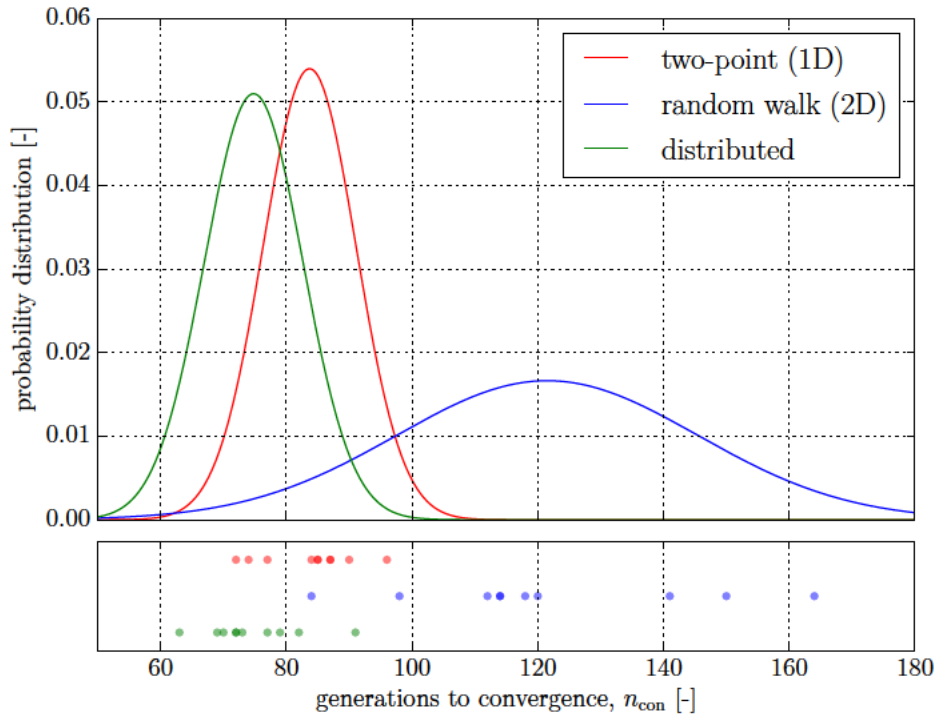
Therefore, we obtain statistical significance that the distributed crossover both converges faster ($j = 1$ in Table 3.6) and yields better final designs ($j = 2$ in Table 3.6) than the random walk crossover. In addition, the two-point crossover also converges faster than the random walk crossover ($j = 3$ in Table 3.6). The result is against the indication, found in the literature (Section 2.3.1), that two-dimensional encoding, due to the better geographical linkage of elements, would provide better results in design problems with a two-dimensional architecture.

Further, we obtain statistical significance that the distributed crossover convergences even faster than the two-point crossover ($j = 4$ in Table 3.6). This indicates that in the current application the diversity of splicing strategies a crossover operator can produce is more important than the geographical linkage between the elements of a genotype.

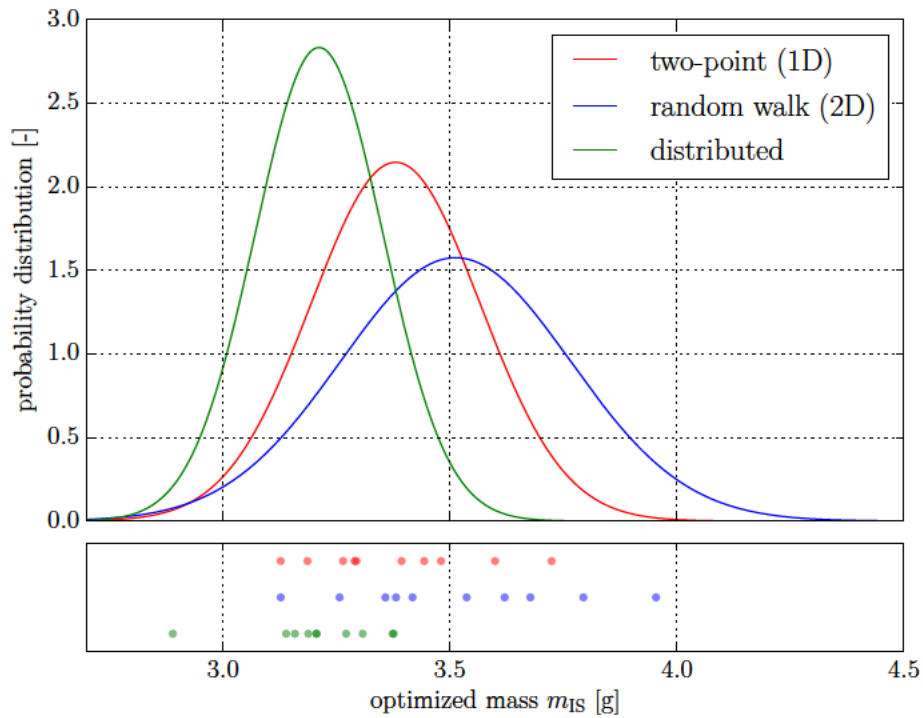
As a conclusion, the distributed crossover has the best performance of the tested crossovers in the current application (with the other selected optimization parameters).

Rank j	Quantity	Crossover 1	Crossover 2	U	p	$j\alpha_s/n$
1	n_{con}	Distributed	Random walk	1.0	0.000243	0.00833
2	n_{con}	Two-point	Random walk	6.5	0.00113	0.0167
3	m_{IS}	Distributed	Random walk	15.0	0.00908	0.0250
4	n_{con}	Distributed	Two-point	19.5	0.0230	0.0333
5	m_{IS}	Distributed	Two-point	26.0	0.0756	0.0417
6	m_{IS}	Two-point	Random walk	35.5	0.290	0.0500

Table 3.6: The family of statistical tests ($j = 1, 2, \dots, 6$) ranked in decreasing order of significance. In each test, Crossover 1 has on average of the obtained results a better performance than Crossover 2. Based on the results, the null hypothesis H_0 is rejected in tests $j = 1, 2, \dots, 4$ and accepted in tests $j = 5, 6$.



(a)



(b)

Figure 3.17: Statistical results of the required generations to converge (a) and the optimized mass (b) on the tree experimented crossovers, and the corresponding normal distribution. Optimization runs with each crossover are repeated 10 times.

3.7.4.5 (B)ESO-based optimization parameters

In this section, we study three optimization parameters related to the (B)ESO-based optimization method. The first is whether structural member recoveries are allowed (ESO/BESO). The second is the number of rejected structural members per iteration, N^{rej} . In addition, when using BESO, the third is the maximum number of recovered structural members per iteration, N^{rec} . We have conducted five optimization runs to study the effects of these parameters (Table 3.7). Since (B)ESO is a deterministic optimization method, the optimization runs were not repeated.

run ID	(B)ESO	N^{rej}	N^{rec}
1	ESO	iter. 0-10: 10 iter. 10-20: 5 iter. 20-: 3	N/A
2	BESO	iter. 0-10: 10 iter. 10-20: 5 iter. 20-: 3	unlimited
3	BESO	iter. 0-10: 10 iter. 10-20: 5 iter. 20-: 3	2
4	ESO	iter. 0-15: 6 iter. 15-30: 3 iter. 30-50: 2 iter. 50-: 1	N/A
5	BESO	iter. 0-15: 6 iter. 15-30: 3 iter. 30-50: 2 iter. 50-: 1	1

Table 3.7: Parameter combinations of the executed (B)ESO-based optimization runs.

The number of rejected structural members is defined to decrease as a function of the iteration number. The reason is that the sensitivities of the structural members are tested individually without actually knowing the combined sensitivities of a set of structural members. This is not critical at the beginning of the process, where several structural members can be removed at the same iteration without significantly decreasing the critical buckling load. However, towards the end of the process, the structure becomes more sensitive and the risk of rejecting a set of structural members with a significant combined sensitivity increases.

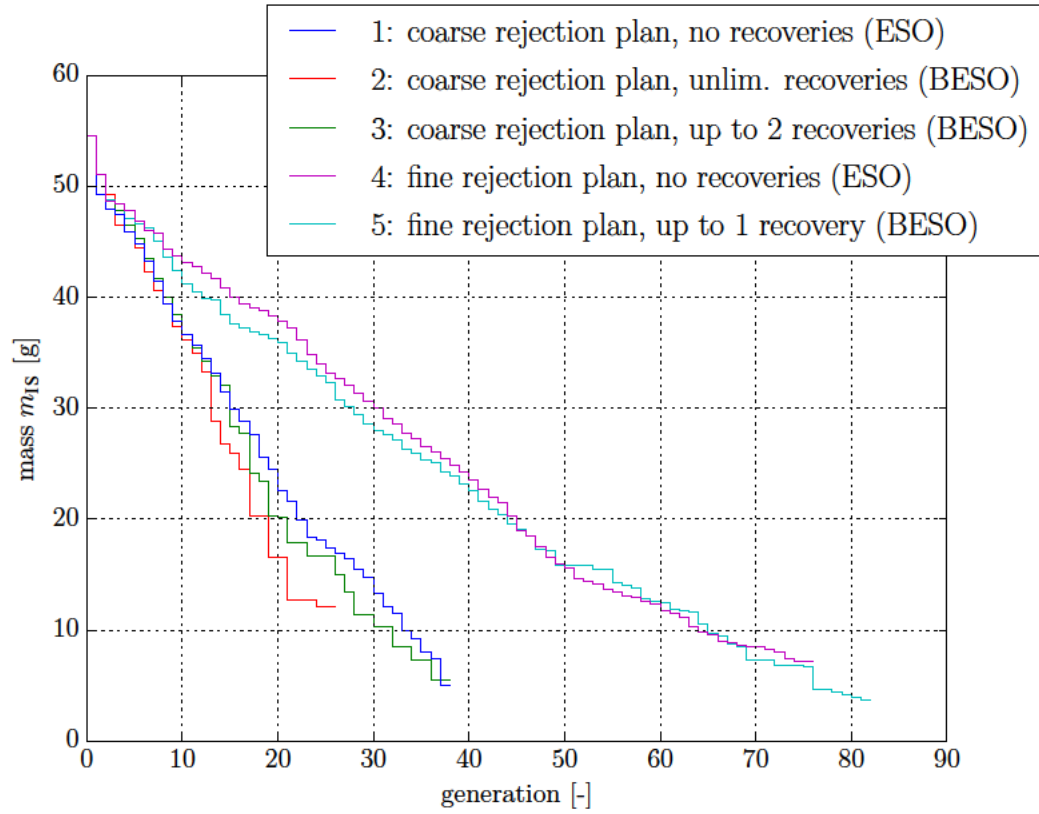
Figure 3.18 presents the evolution of the internal structure mass (Subfigure a) and critical buckling load (Subfigure b) as a function of the iteration number. Since the same ground structure (GS1) is used in all optimization runs, they all start from the same structural mass and critical buckling load. During the first iterations, the critical buckling load actually increases, which seems to run contrary to intuition. We believe

that the reason is in the definition of the member option hierarchy. At the beginning of the rejection process, structural members are changed from full-depth members to stiffeners on the upper skin, which in some parts of the structure seem to provide better resistance against buckling.

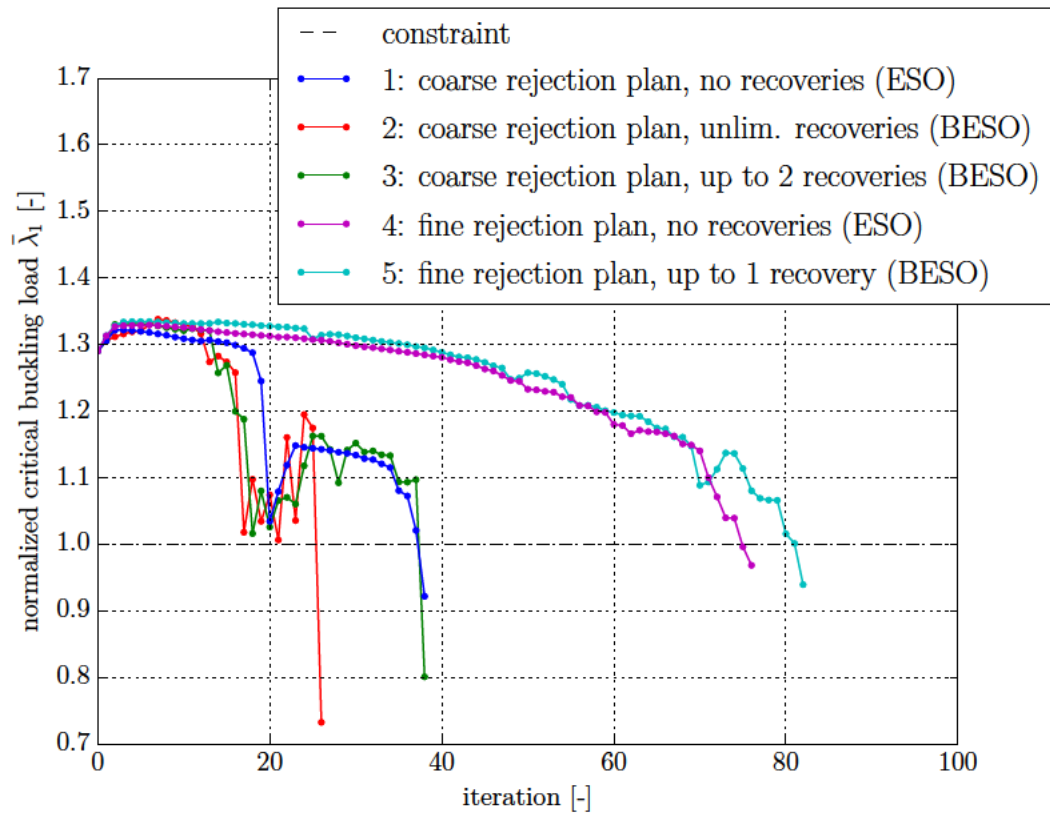
Optimization runs 1 and 2 were executed first to compare the performances of ESO and BESO-based optimization methods with a fairly coarse rejection plan (see Table 3.7). As it can be seen from Figure 3.18(b), the normalized critical buckling load decreases rapidly at around iterations 15-18, indicating that too many structural members have been rejected from the same region during an iteration. Optimization run 2 uses the BESO definition with an unlimited number of structural member recoveries. As a consequence, the rapid decrease in the critical buckling load triggers an oscillation phenomenon, where several members are moved back and forth between two regions of the wing on consecutive iterations. Eventually, the oscillation causes the termination of the process when the normalized critical buckling load becomes less than unity.

To avoid the oscillation, we have limited the maximum number of structural member recoveries per iteration, N^{rec} , to two in run 3. As we can see in the figure, this run has a more stable behavior than optimization run 2. Meanwhile, the normalized critical buckling load of optimization run 1, where no recoveries are made, increases steadily from 1.03 to 1.15. In a similar fashion to that seen at the beginning of the optimization process, several full-depth structural members are rejected to stiffeners on the upper skin during these iterations. However, the behavior is not fully understood. With the coarse rejection plan, optimization runs with ESO- (run 1) and BESO-based (run 3) methods yield similar results.

Next, ESO- (run 4) and BESO-based (run 5) optimizations were performed with a finer rejection plan. This time no rapid decrease in the critical buckling load was observed during the optimization. For most of the optimization process, the BESO-based optimization has slightly greater critical buckling load than the ESO-based optimization. In contrast to the coarse rejection plan, this time the BESO-based optimization method yields a design that is 3.55 grams lighter than that produced by the ESO-based optimization method.



(a)



(b)

Figure 3.18: Variation of the internal structure mass (a) and the critical buckling load (b) during (B)ESO-based optimization runs on GS1.

3.7.4.6 Choosing the parameters

So far in this chapter, the focus has been on finding reasonable main parameters for the two optimization methods. The studied parameters were the crossover type and the population size in the GA and the rejection plan and the bi-directionality in (B)ESO. In the following two sections (Section 3.7.4.7 and 3.7.4.8), we apply the methods to the remaining ground structures, presented in Section 3.7.3. In this section, we choose appropriate values for the main parameters, using the knowledge obtained from the parameter studies.

With the GA, we came to a conclusion that, for the current application, the distributed crossover has the best performance out of the three tested types of crossover operators. We also briefly studied the effect of the population size N_{pop} ; no major difference was observed in the optimized mass with population sizes 150 and 300. Smaller population sizes, 38 and 75, were observed, in some cases, to converge prematurely to a local minimum. However, the sample size was only three (due to the high cost of the runs), so more experiments should be performed for statistically significant conclusions. In the remainder of this chapter, we choose the crossover type to be the distributed crossover and the population size to be 150 (with some exceptions that are indicated in the text).

With (B)ESO heuristic, we discovered that a coarse rejection plan (runs 1, 2 and 3 in Table 3.7) may reject multiple ground structure members from the same region in the wing in a single iteration, what exposes the optimization process to a premature termination. This behavior is avoided by a finer rejection plan (runs 4 and 5 in Table 3.7). To be able to apply this result to other ground structures (with different numbers of ground structure slots), we define a rejection plan that is a function of the number of rejections n that can be performed from the remaining ground structure. This rejection plan is defined as

$$N^{\text{rej}} = \begin{cases} 6, & \text{if } n > 0.6n_{\text{tot}} \\ 3, & \text{if } 0.6n_{\text{tot}} > n > 0.4n_{\text{tot}} \\ 2, & \text{if } 0.4n_{\text{tot}} > n > 0.25n_{\text{tot}} \\ 1, & \text{if } 0.25n_{\text{tot}} > n \end{cases} \quad (3.10)$$

where n_{tot} is the number of possible recoveries from the full ground structure. This rejection plan is equivalent to the experimented finer rejection plan on GS1. We use it in the remaining of this chapter with the (B)ESO heuristics.

Finally, it is worth noting that the parameter studies are performed on GS1, which has the smallest number of member slots out of the described ground structures in Section 3.7.3. The chosen parameters might not be ideal for other ground structure types or densities. While the number of member slots in GS1 is 110, the densest version of GS3 has a total of 295 member slots. With that many member slots, performing statistical tests is not feasible with our current computation resources. The parameters are chosen using our best knowledge obtained from the results with GS1.

3.7.4.7 Quadrilateral (GS1) and diagonal (GS2) ground structures

In the literature review, we found indications that including diagonal members in the internal structure would improve the total performance of the aircraft wing structure (Section 2.8.1.1). This section examines the obtained results with GS1 and GS2, from which GS1 has only span- and chordwise ground structure slots and GS2, in addition to the previous, has the diagonal slots. In addition, we examine the differences between the results obtained using GAs and (B)ESO heuristics.

Let us start by examining the evolution process in GA-, ESO- and BESO-based methods. Figure 3.19 presents representative individuals of optimization runs performed on GS1. The individuals of the GA-based optimization are generation bests of the same optimization as in Figures 3.13 and 3.14. The individuals of the ESO- and BESO-based optimizations correspond to runs 4 and 5 in Table 3.7. While the ESO- and BESO-based methods are initiated from the full ground structure, the GA-based method initiated by randomly generating a population of 150 individuals.

All optimized designs (Subfigures 3.19(m) to 3.19(o)) have several continuous lines of longitudinal stiffeners (on the upper skin) starting from the root. Out of these stiffener lines, the longest lines are located in between the mid-chord and trailing edge. In this region, the curvature of the upper skin is the smallest, and thus the most prone to buckling. In addition, the designs all have two to five full-depth members connecting the two skins. These full-depth structural members also provide buckling resistance for the lower skin, which is prone to a snap-through buckling.

The numerical values of the optimized masses are listed in Table 3.8. The value for GS1 with the GA is an average of 10 repeated optimizations with the distributed crossover. Earlier, we observed that the use of the bi-directional feature of ESO (BESO) improved the optimized mass by 3.55 grams (Section 3.7.4.5). As we can see in Subfigures 3.19(m) to 3.19(o), the main line of full-depth structural members lies in the design resulting from ESO at around the mid-chord, whereas in the design resulting from BESO it is closer to the trailing edge. The location near the trailing edge not only provides more support for the section most prone for buckling but also enables the connection of the two skin with less material, due to the smaller profile thickness.

The optimized designs initiated from the diagonal ground structure (GS2) are presented in 3.20. While the dominant structural members are still the spanwise lines of stiffeners, the designs also contain three to four diagonally orientated stiffeners or full-depth members. The design obtained with the ESO-based method has a lower optimized mass ($m_{IS} = 2.73$ g) than the corresponding design from GS1 ($m_{IS} = 7.24$ g). On the other hand, the design obtained with the GA-based method has an optimized mass ($m_{IS} = 3.24$ g) similar to the corresponding average of designs from GS1 ($m_{IS} = 3.21$ g). However, it should be noted, again, that GAs are stochastic optimization methods, and we have

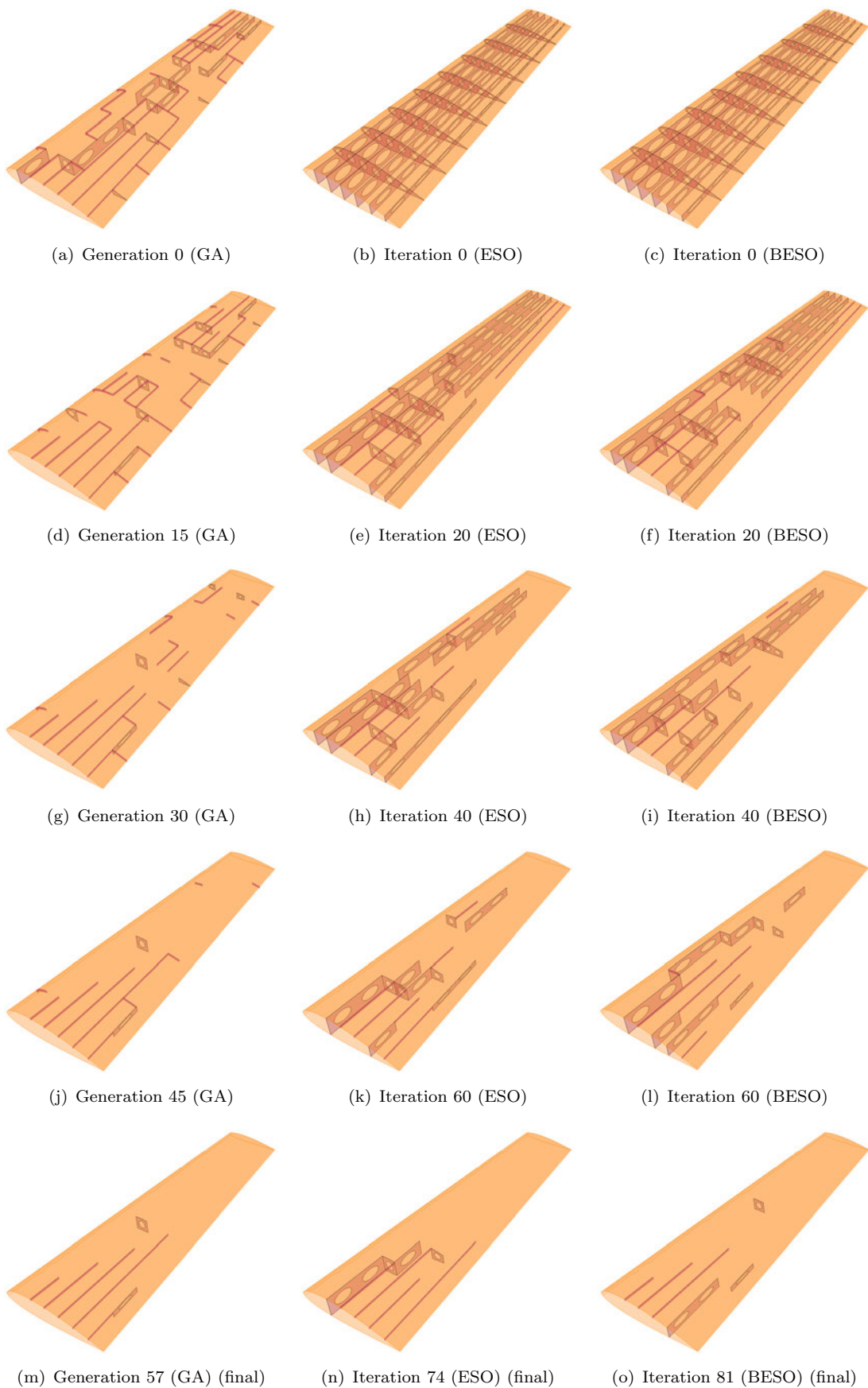


Figure 3.19: Evolution of the internal structure using the GA-, ESO- and BESO-based optimization methods on GS1.

Ground structure	Optimized mass m_{IS} [g]		
	GA	ESO	BESO
GS1	3.21 ± 0.141^i	7.24	3.68
GS2	3.24^{ii}	2.73	3.96

ⁱ Average and standard deviation of 10 repeated optimizations with the distributed crossover. ⁱⁱ 2.96, if $N_{pop} = 300$.

Table 3.8: Optimized masses of quadrilateral (GS1) and diagonal (GS2) ground structures using GA-, ESO- and BESO-based methods.

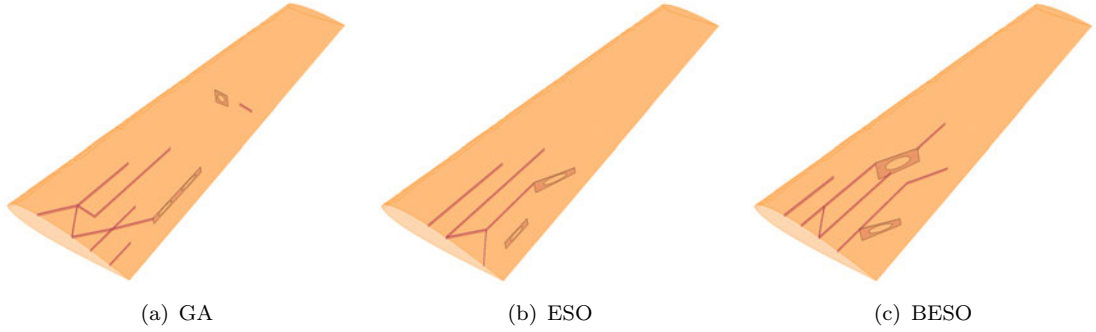


Figure 3.20: Optimized structures from optimization runs initiated from the diagonal ground structure (GS2). Results with GA, ESO and BESO are presented.

only a sample size of one on GS2. The GA-based optimization was repeated with a population size of $N_{pop} = 300$, yielding 9.5% lighter design.

3.7.4.8 Hexagonal ground structures (GS3)

We have earlier discussed that the quadrilateral and diagonal ground structures may have a structural weakness, since they allow the formation of T-junctions in the internal structure (Section 3.3). The T-junctions are avoided with the hexagonal ground structure (GS3), which we discuss here. In addition, we examine the effect of the ground structure density on the optimized design. To reduce the computational cost, the BESO-based optimization runs were executed without the bi-directional feature until 20% of the total number of rejections were remaining.

A total of 12 optimization runs were performed with GS3. The results form a four-by-three matrix where the rows correspond to the densities of the ground structure and the columns to the optimization method. The optimized masses are presented in Table 3.9 and optimized designs in Figure 3.21. Figure 3.22 plots the structural masses with different densities, along earlier obtained results with GS1 and GS2.

Based on the results, optimizations initiated from Density 1 clearly have worse final designs than the optimization initiated from denser ground structures. As far as we

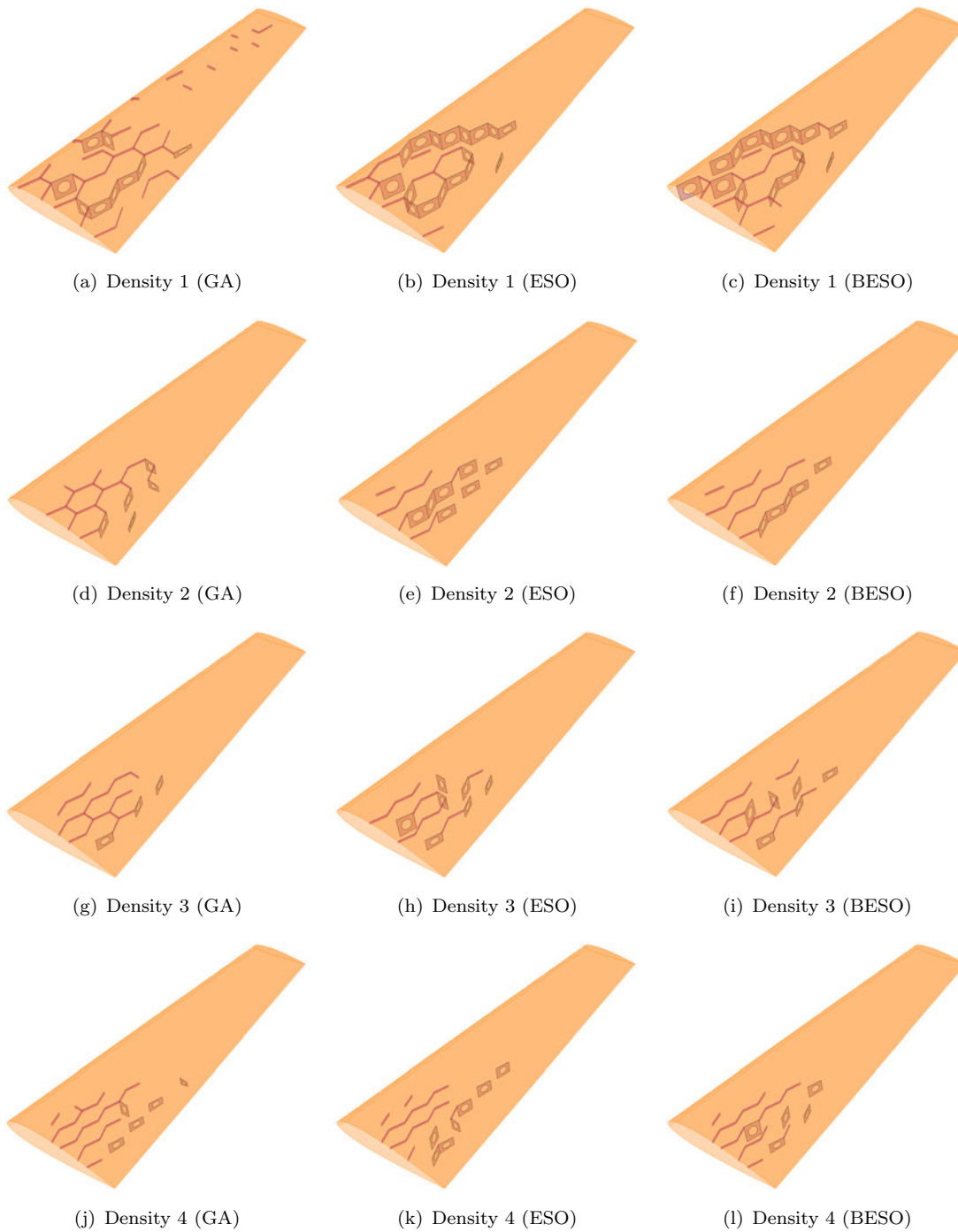


Figure 3.21: Optimized structures from optimization runs initiated from the four different ground structure densities of the hexagonal ground structure (GS3).

Ground structure	Optimized mass m_{IS} [g]		
	GA	ESO	BESO
GS3: density 1	7.82	9.28 ⁱ	9.53 ⁱ
GS3: density 2	2.97	3.99	3.04
GS3: density 3	2.59	3.65	3.49
GS3: density 4	2.63	3.26	3.07

ⁱ Modified rejection plan: $N^{\text{rej}} = 4$, if $n > 0.6n_{\text{tot}}$, else 2, if $0.6n_{\text{tot}} > n > 0.4n_{\text{tot}}$, else 1 (see Eq. 3.10)

Table 3.9: Optimized masses of the hexagonal ground structures (GS3) with different densities using GA-, ESO- and BESO-based methods.

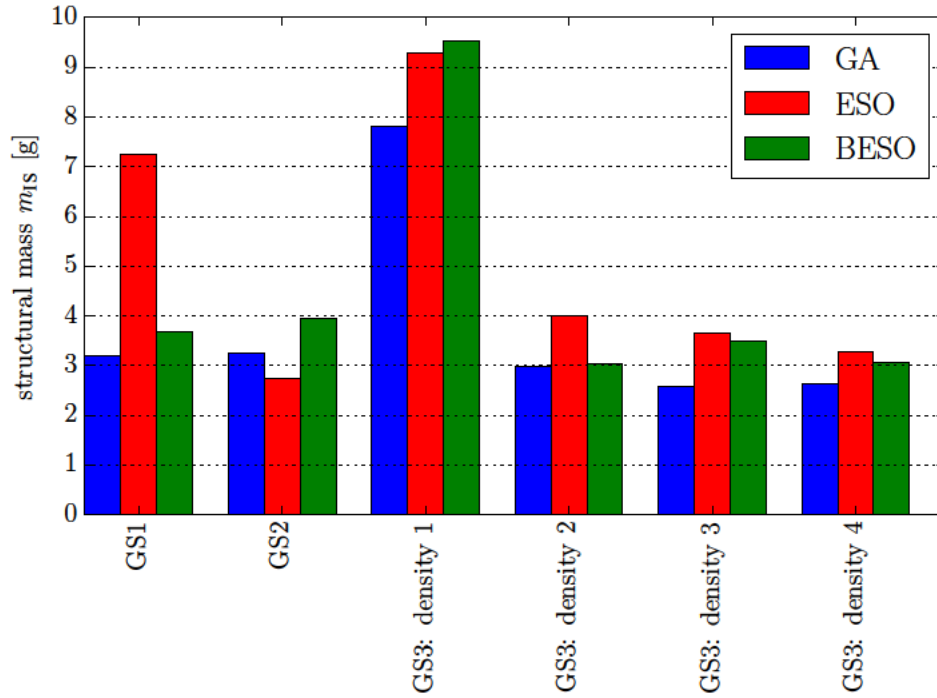


Figure 3.22: Optimized internal structure masses from GS1, GS2 and GS3 with four ground structure densities.

know, the reason is the following. Even if all member slots in the ground structure are filled with full-depth members, the normalized critical buckling load has a value of $\bar{\lambda}_1 = 1.03$, which is only barely above the constraint. Rejection of any member in the structure may result in the normalized critical buckling load decreasing below the constraint. In fact, the results presented here with (B)ESO heuristics were performed with an even finer rejection plan, but the optimized masses are still clearly worse than with the denser ground structures. The GA-based method can find a slightly better design, but also struggles with the narrow feasible design region.

In general, optimization runs on the denser ground structures yield better designs. These designs (Figures 3.21(d)-3.21(l)) have, in a similar fashion to those obtained from GS1 and GS2, a number of stiffener members near the root of the wing and three to seven full-depth structural members close to the trailing edge. Some of the designs, such as Density 2 with BESO heuristics (Figure 3.21(f)) and Density 4 with ESO heuristics (Figure 3.21(k)), have zigzagging stiffener lines, somewhat similar to those obtained from GS1. Based on Table 3.9, the GA-based method found designs with lower structural mass than (B)ESO heuristics with all ground structure densities. The lightest design ($m_{\text{IS}} = 2.59 \text{ g}$) from all tested ground structures was obtained with the GA-based method from GS3 with Density 3.

Despite the fact that the designs, obtained from all ground structures, have similar features, they are clearly not identical. Thus, the current optimization problem has a multi-modal landscape. Considering that GS2, as an example, has 173 ground structure slots and each slot has three different options, the total of different subsets of options drawn from GS2 is $3^{173} \approx 3.48 \times 10^{82}$. Finding the global optimum from this large design space is nearly impossible. However, finding a good local optimum is often sufficient for practical design tasks. In the next section, we compare these local optima to the earlier presented manual designs.

3.7.4.9 Comparison to conventional design methods

In Section 3.7.2, we presented conventional solutions to the same design problem. The best conventional design, i.e. the stiffener design (Figure 3.9(b)), is outperformed by topology optimization runs initiated from all ground structures, except the ones from GS3 with Density 1, regardless of the optimization method. The lightest obtained design from the ESO-based optimization method with GS3 (Density 3) is 64% lighter than the stiffener design. The design is feasible, i.e. it does not violate any constraints specified in Section 3.7.1.

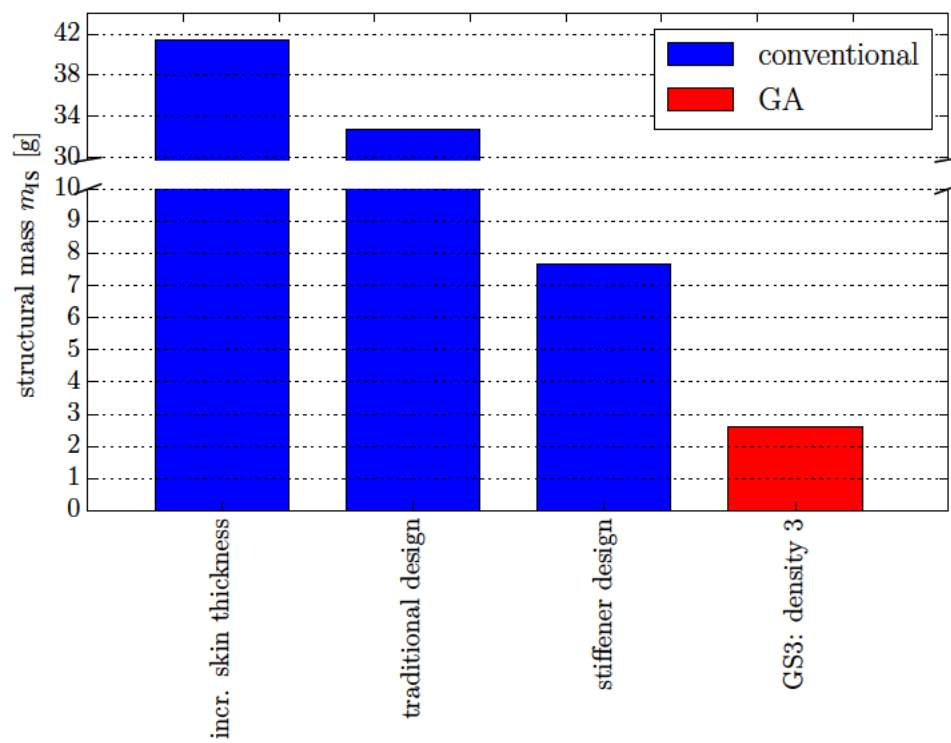


Figure 3.23: Comparison of the lightest obtained design from topology optimization against the conventional design methods.

3.8 Conclusions

In this chapter, we presented two ground structure approaches, based on a GA and (B)ESO heuristics, to topology optimization of the aircraft wing internal structure. The methods were applied to a design problem of a sUAV wing, built via additive manufacturing.

Earlier in Section 2.3.1.1, we found indications in the literature that GAs with two-dimensional encoding perform well on optimization problems with a two-dimensional architecture. Thus, in this chapter, we performed a statistical experiment of crossover types, which use one- and two-dimensional encodings. The studied crossover types were two-point crossover in one-dimensional encoding, random walk in two-dimensional encoding, and distributed crossover. However, contrary to the indications found in the literature, we obtain a statistically significant result that, on the presented optimization problem, the random walk crossover converges slower than two-point and distributed crossovers, and has worse optimized designs than the distributed crossover. Further, the distributed crossover provides better optimized designs than the two-point crossover, which highlights the importance of diverse splicing strategies that a crossover operator is capable of producing. Since the statistical tests were conducted as a family, the inflated false discovery rate was adjusted using the step-up FDR. Therefore, on average, five percent of the discovered statistically significant results are false discoveries.

The (B)ESO-based approach was found to be critical to the number of rejected ground structure members per iteration. If multiple ground structure members are rejected from the same area in a single iteration, the critical buckling load might drop under termination criterion and the process is terminated prematurely. Using a relevant rejection plan, bi-directional ESO (BESO) was demonstrated to outperform ESO.

The resulting designs obtained from GA- and (B)ESO-based approaches outperform the presented manual design methods. The obtained designs do not have the familiar spar-rib arrangements, typically seen in conventional aircraft structures, which raises interesting questions in terms of wing structural design in general. However, it is worth mentioning that the prevailing, relatively high, minimum wall thickness constraint reduces the role of the internal structure.

A limitation with the ground structure approach is that the discrete ground structure must be defined a priori, which both requires an additional step in the optimization process and narrows the design space. In the next two chapters, we study the map L-systems-based method, where the designs are constructed implicitly via recipes that mimic the behavior of the DNA sequence. The method is free of the above mentioned limitations.

Chapter 4

Map L-systems-based method: statistical experiment

In this work, we have narrowed the scope of generative encodings to those based on L-systems, which we reviewed in Section 2.7. L-systems mimic the developmental process of living organisms. When used as a parameterization method¹, they are described to be able to cover a strikingly diverse design space while still using relatively few design variables (Deaton and Grandhi, 2014). We reviewed two interpretation formalisms L-systems, i.e. turtle interpretation and map L-systems, in Sections 2.7.1 and 2.7.2, respectively.

Recently, especially the map L-systems-based parameterization has gained popularity among topology optimization researchers (see Section 2.7.4). In the majority of published papers, the map L-systems-based parameterization is evolved by genetic algorithms (GAs). Moreover, several authors, e.g. Pedro and Kobayashi (2011), Stanford et al. (2012, 2013) and Allison et al. (2013), used similar numerical representations of map L-systems, which originate from that defined by Pedro and Kobayashi (2011) (see Section 2.7.3). Despite of the number of studies with similar algorithms and numerical representations, no systematic efforts have been reported to date to understanding the impact of the control parameter choices of GAs on the performance of these optimization processes.

We acknowledge that, in general, 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, parameterizations where the evolutionary process operates on the encoding – such as L-systems based methods – and not directly on the design, should be less affected by this problem dependence. The encoding can be seen as an intermediary layer of the problem, which ‘shields’ the evolutionary search

¹It is worth noting that, originally, L-systems were not designed to be evolved, but to mathematically represent the topological/geometrical development of living organisms.

from some of the variability resulting from the objective function of the structural design problem.

In this chapter, we perform a statistical experiment involving 432 control parameter combinations on the map L-systems-based topology optimization method, using a numerical representation similar to that proposed by [Pedro and Kobayashi \(2011\)](#). Our experiment involves a significantly large number of objective function evaluations (in an order of one billion). Performing this many time-consuming FE analysis-based objective function (e.g. that used in the previous chapter) evaluations would not be feasible. Thus, in order to keep the computational cost to a minimum and to facilitate the interpretation of the results, we devise five simple optimization problems, which we define using geometric features of the phenotypes. In addition, we define them to have known global optima, to facilitate the evaluation of the optimized designs. In order to distinguish these optimization problems from the applications of this work, we refer to them as the *test cases*.

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.

Before going into the details of the statistical experiment, let us start by defining the map L-systems design space (Section 4.1) and demonstrating the optimization method on one of the test cases (Section 4.2).

4.1 Defining the design space

Here, we make two minor modifications to the numerical representation by [Pedro and Kobayashi \(2011\)](#). First, [Pedro and Kobayashi \(2011\)](#) define the sixth real number of a token to vary a specific property of the edge (e.g. the thickness). The edge property is redundant in our test cases and, therefore, we omit it. Thus, our design variable vector $\mathbf{x}$ has a total length of

$$n_d = N_a + 5N_rN_P + N_v. \quad (4.1)$$

Second, in the encoding by [Pedro and Kobayashi \(2011\)](#), the third element of $\beta_{j,k}$ (Equation 2.27) defines the edge/marker orientation. For simplicity, we define all edges to have a neutral orientation.

The parameters defining the design space via map L-systems are listed in Table 4.1. The axiom of map L-systems is mapped as a unit square, and thus the axiom length $N_a = 4$. We use two additional variables: f_a defines the minimum fraction between offspring and

parent cell areas, and n is the age of the system. These parameters are kept constant throughout the statistical experiment. However, after the statistical experiment, we will test the sensitivity of the obtained results to the number of encoded rewriting rules, N_P .

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 ... 0.5
age n	1 ... 6

Table 4.1: Definition of the L-system design space. Minimum area fraction f_a and age n are additional variables.

The map L-systems could be amended by a dynamic method ([Prusinkiewicz and Lindenmayer, 2012](#)) (Section 2.7.2), 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.

4.2 Introductory example

This section presents an introductory example of evolving the L-systems-based geometry description via a GA². The purpose of the optimization problem presented here is to, first, illustrate the use of map L-systems-based parameterization in geometry optimization and, second, serve as the first test case for the statistical experiment.

4.2.1 Test Case 1

The first test case is inspired by a map L-system, described by [Prusinkiewicz and Lindenmayer \(2012\)](#), in which the axiom and rewriting rules are chosen to be

$$\begin{aligned}
 \text{Axiom: } \omega_0 &= ABAB \\
 \text{Rules: } P_1 : A &\rightarrow B[-A][+A]B \\
 P_2 : B &\rightarrow A
 \end{aligned} \tag{4.2}$$

When the rewriting rules are applied four times, the obtained map includes 16 equally sized cells, which all have a square shape (Figure 4.1). In fact, every odd developmental

²We implement our GAs also in this chapter using Pyevolve ([Perone, 2009](#)), an open source Python library of evolutionary operators.

stage of this system is a map including $N_{\text{cells}} = n^2$ equally-sized, square-shaped cells, where n is age of the system.

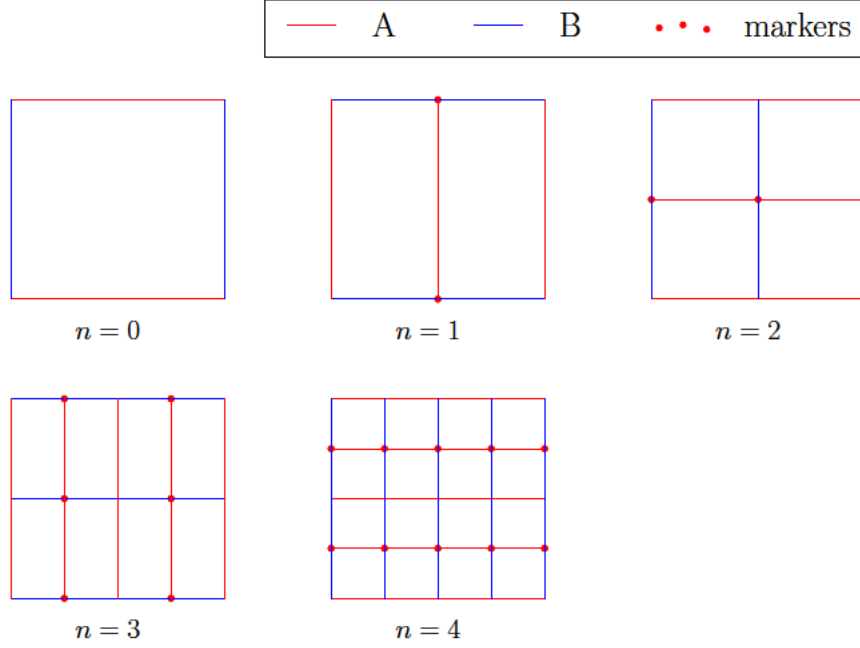


Figure 4.1: A map L-system producing a square grid.

The goal in this test case is to find an axiom, rewriting rules, and additional variables that produce a map consisting of 16 equally-sized, square-shaped cells. To obtain a design landscape where the desired map is the global optimum, we penalize the following quantities of the map: the absolute difference of the number of cells to the desired number of cells, $|16 - N_{\text{cells}}|$, the standard deviation of the edge lengths S_e , and the standard deviation of the cell areas S_a . Thus, we define the objective function as

$$f_1 = \frac{1}{10}|16 - N_{\text{cells}}| + S_e + S_a, \quad (4.3)$$

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 map, respectively.

Although the map L-system described in Equation 4.2 produces the desired map with $n = 4$, it is not necessarily the only system for success. In terms of the map, the defined objective function only has a single global optimum, however, there may be multiple L-systems producing the same map.

4.2.2 Results

We deploy a GA³, with the numerical representation described in Sections 2.7.3 and 4.1, to tackle this introductory optimization problem. Figure 4.2 shows two representative optimization runs. The first yields the global optimum of $f_1 = 0$, whereas the optimized design of the second is sub-optimal.

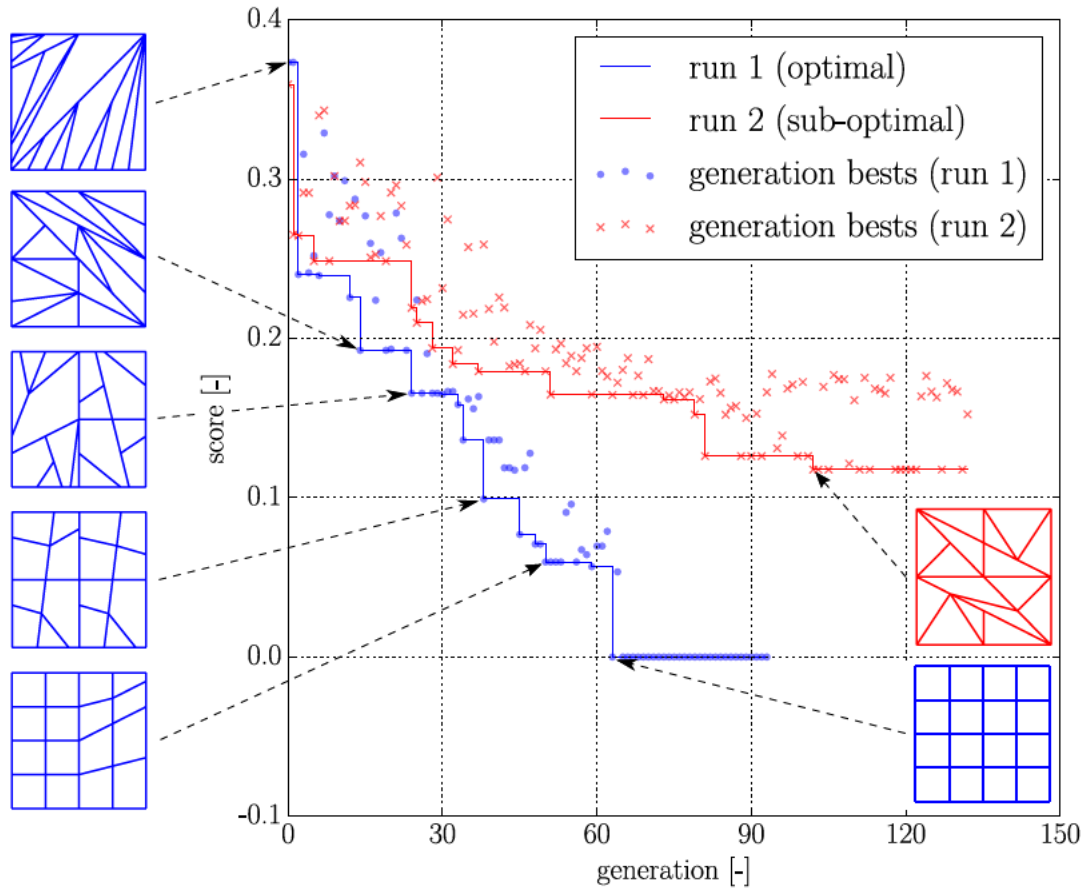


Figure 4.2: Representative optimization runs yielding the global optimum and a sub-optimal final result.

³The control parameters used here are the same as in the previous chapter (with distributed crossover and $N_{\text{pop}} = 150$)

The corresponding map L-system of the optimal design, obtained from the first run, is the following

$$\begin{aligned}
\text{Axiom:} & \quad \omega_0 = ABAD \\
\text{Rules:} & \quad P_1 : A \rightarrow B[-A][+A][+A]B \\
& \quad P_2 : B \rightarrow [-\overleftarrow{D}][+\overrightarrow{C}][+\overleftarrow{B}]A[-\overleftarrow{B}] \\
& \quad P_3 : C \rightarrow AB[-D]CC \\
& \quad P_4 : D \rightarrow [+C][+\overrightarrow{C}][-\overleftarrow{B}][+A]A \\
\text{Additional properties:} & \quad n = 4 \\
& \quad f_a = 0.3192
\end{aligned} \tag{4.4}$$

As we can see, the age of this system is the same ($n = 4$) as in the reference system (Equation 4.2). Further, the rewriting rules P_1 and P_2 , despite having additional (inactive) markers, correspond to the rewriting rule P_1 and P_2 , respectively, in the reference system. The last letter of the axiom in the obtained system is D , while in the reference system it is B . However, as the successors of rewriting rules P_2 and P_4 in the obtained system are equivalent, the system yields the same phenotype as the reference system.

In the following, we describe the experimental plan (Section 4.3) and the remaining four test cases (Section 4.4). Finally, we deploy these to gain an empirical understanding of the performance of the algorithm with a range of control parameter choices (Section 4.5).

4.3 Experimental plan

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).

Table 4.2 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.

Publication	Selection	Population size N_{pop}	Crossover	Mutation	Elitism E_{bool}	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) ⁱ	True	terminated after 500 generations
Kobayashi (2010) ⁱⁱ	tournament ($N_{\text{pool}} = 4$)	200	distributed ($c_x = 0.8$)	Gaussian distributed ⁱⁱⁱ ($c_M = 0.2$)	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) ⁱⁱ	tournament ($N_{\text{pool}} = 4$)	100	distributed ($c_x = 0.8$)	Gaussian distributed ⁱⁱⁱ ($c_M = 0.19$)	True	terminated after 100 generations
Ikonen and Sóbestor (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

ⁱ 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.

ⁱⁱ In addition, the author(s) applies inversion to the axiom letters, with a probability of 0.01. ⁱⁱⁱ 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.

Table 4.2: Survey of control parameters in L-systems-based topology optimization studies found in the literature.

Mutation rate c_m is the element-specific probability of mutation, applied to a c_M proportion of the population. In the studies by Kobayashi (2010) and Sabbatini et al. (2015) the generation of phenotype follows the turtle interpretation. 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 4.2 shows, the variation in control 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 run a statistical experiment on the five test cases. Table 4.3 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 test all 432 control parameter combinations 70 times on each of the five test cases. The optimization runs are terminated when no improvements were obtained during 30 consecutive generations. We use 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 are initiated from a population of random individuals.

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_{bool}	{True, False}

Table 4.3: Control parameter values of the statistical experiment. All parameter combinations, totalling 432, are tested separately.

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 the paper by Simpson and Dsouza (2004)). However, in the vast majority of studies, using an L-systems-based parameterization, optimization runs are initiated from a random population⁴. 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 work.

⁴An 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.

4.4 Remaining test cases

The first test case was defined as a part of the introductory example in Section 4.2.1. This section defines the remaining four test cases.

4.4.1 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

$$f_2 = \frac{1}{2}P_N + 2 \frac{\sum_{i=1}^{N_{\text{cells}}} s_{e,i}}{N_{\text{cells}}} + \frac{1}{100} \frac{\sum_{i=1}^{N_{\text{cells}}} s_{\alpha,i}}{N_{\text{cells}}}, \quad (4.5)$$

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

$$P_N = \begin{cases} 5 - N_{\text{cells}}, & \text{if } N_{\text{cells}} < 5 \\ 0, & \text{if } N_{\text{cells}} \geq 5 \wedge N_{\text{cells}} \neq 16 \\ 1, & \text{if } N_{\text{cells}} = 16, \end{cases} \quad (4.6)$$

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 Figure 4.3(a).

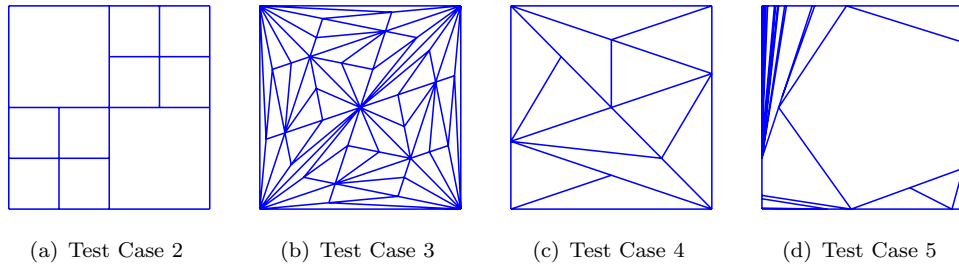


Figure 4.3: Example global optima in Test Cases 2-4, and the design with the lowest objective function value in Test Case 5. The map L-systems of these phenotypes are listed in Appendix B.

4.4.2 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

$$f_3 = \frac{N_{\text{nodes}}}{N_{\text{cells}}}. \quad (4.7)$$

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

$$N_{\text{cells}} = 2N_{\text{nodes}} - B_{\text{nodes}} - 2, \quad (4.8)$$

where B_{nodes} is the number of nodes laying at the convex boundary of the graph⁵. The equation can be rewritten as

$$\frac{N_{\text{nodes}}}{N_{\text{cells}}} = \frac{1}{2} + \frac{B_{\text{nodes}} + 2}{2N_{\text{cells}}}. \quad (4.9)$$

Therefore, the objective function f_3 (Equation 4.7) 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 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 Figure 4.3(b). It is noticeable that the boundary of the map only includes the four nodes related to the axiom.

4.4.3 Test Case 4

An *N-equidissection* of a polygon is set of N 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

$$f_4 = \frac{1}{2}|12 - N_{\text{cells}}| + 10S_a + \frac{N_{\text{cells}} - \hat{N}_{\text{cells}}}{N_{\text{cells}}}, \quad (4.10)$$

where $\hat{N}_{\text{cells}}$ and S_a are the number of triangular cells and the standard deviation of cell areas in the map, respectively. An example global optimum ($f_4 = 0$) shown in Figure 4.3(c).

⁵All 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.

4.4.4 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

$$f_5 = P_A + \frac{1}{100}s_{\alpha,k} + s_{e,k}, \quad (4.11)$$

else $f_5 = 10$. P_A is a penalty coefficient defined as

$$P_A = \begin{cases} \frac{1}{4} - A_k, & \text{if } A_k < \frac{1}{4} \\ 0, & \text{if } A_k \geq \frac{1}{4}, \end{cases} \quad (4.12)$$

where A_k is the area of the largest pentagonal shaped cell k . Further, $s_{\alpha,k}$ and $s_{e,k}$ 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 Figure 4.3(d).

4.5 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 4.3, are applied to Test Cases 1-5, each run 70 times. Global optima were found for Test Cases 1-4 (see Figures 4.2 and 4.3(a)-4.3(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 (Figure 4.3(d)) 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 (Figure 4.4) of the average number of objective function 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.

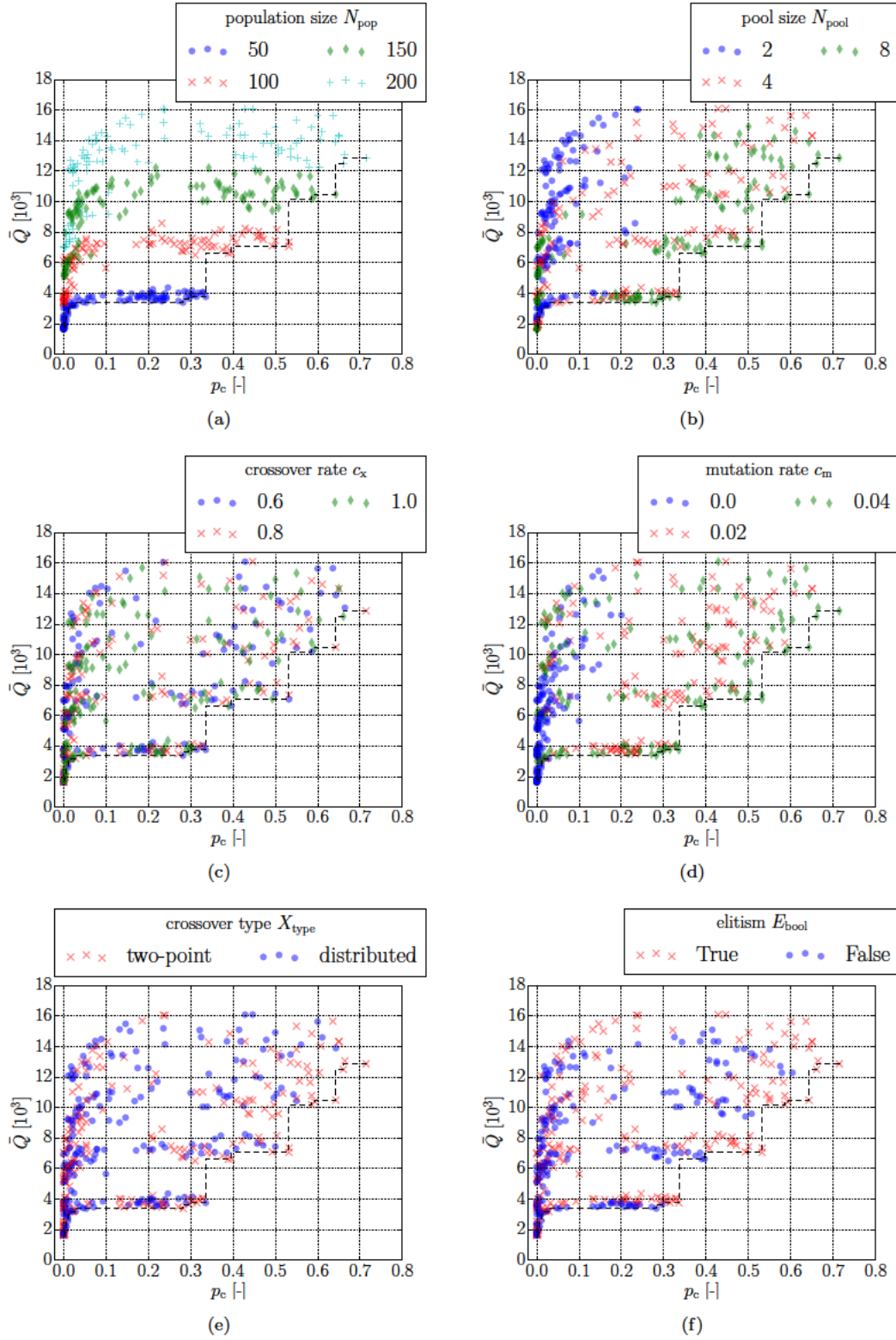


Figure 4.4: Results of the statistical experiment, in terms of the completion rate p_c and the average number of required objective function evaluations $\bar{Q}$. Test Case 5 is excluded from the results, as its global optimum was not found. Each of Subfigures (a)-(f) visualizes the effect of one control parameter on the performance of the algorithm. The dashed line represents the Pareto front between minimum average number of function evaluations and maximum completion rate.

The population size N_{pop} has the clearest influence on the performance of the GA (Figure 4.4(a)). 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} (Figure 4.4(b)) and the element-wise mutation rate c_m (Figure 4.4(d)) 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} (Figure 4.4(f)) divide the four bands of population sizes each into two subbands, again approximately parallel to the abscissa of the plot. The value $E_{\text{bool}} = \text{True}$, 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 (Figure 4.4(e)), and its points form most of the Pareto front. However, the performance difference between the crossovers is small. The crossover rate c_x (Figure 4.4(c)) 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 4.5 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 (Figure 4.4), 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 4.4. 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 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

$$p_{\text{global}} = 1 - (1 - p_c)^{n_{\text{runs}}}, \quad (4.13)$$

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

$$Q_g = \bar{Q}n_{\text{runs}}, \quad (4.14)$$

where $n_{\text{runs}} = \log_{(1-p_c)}(0.05)$, rounded to the next natural number. The smallest $Q_g (= 29.12 \cdot 10^3)$ is obtained by combination #7 in Table 4.4, 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 guidelines for formulated in the general GA literature. However, the tournament pool size $N_{\text{pool}} = 8$ and mutation rate $c_m = 0.04$ 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 4.4 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_s (Corder and Foreman, 2014), which compares the relationship of ordinal or rank-ordered variables. If $\rho_s = 1$, the correlation is perfect, i.e. the parameter combination ranks are the same among the two test cases. If $\rho_s = -1$, the correlation is also perfect but the ranks are the opposite. On the other hand, if $\rho_s = 0$, the ranks are completely independent.

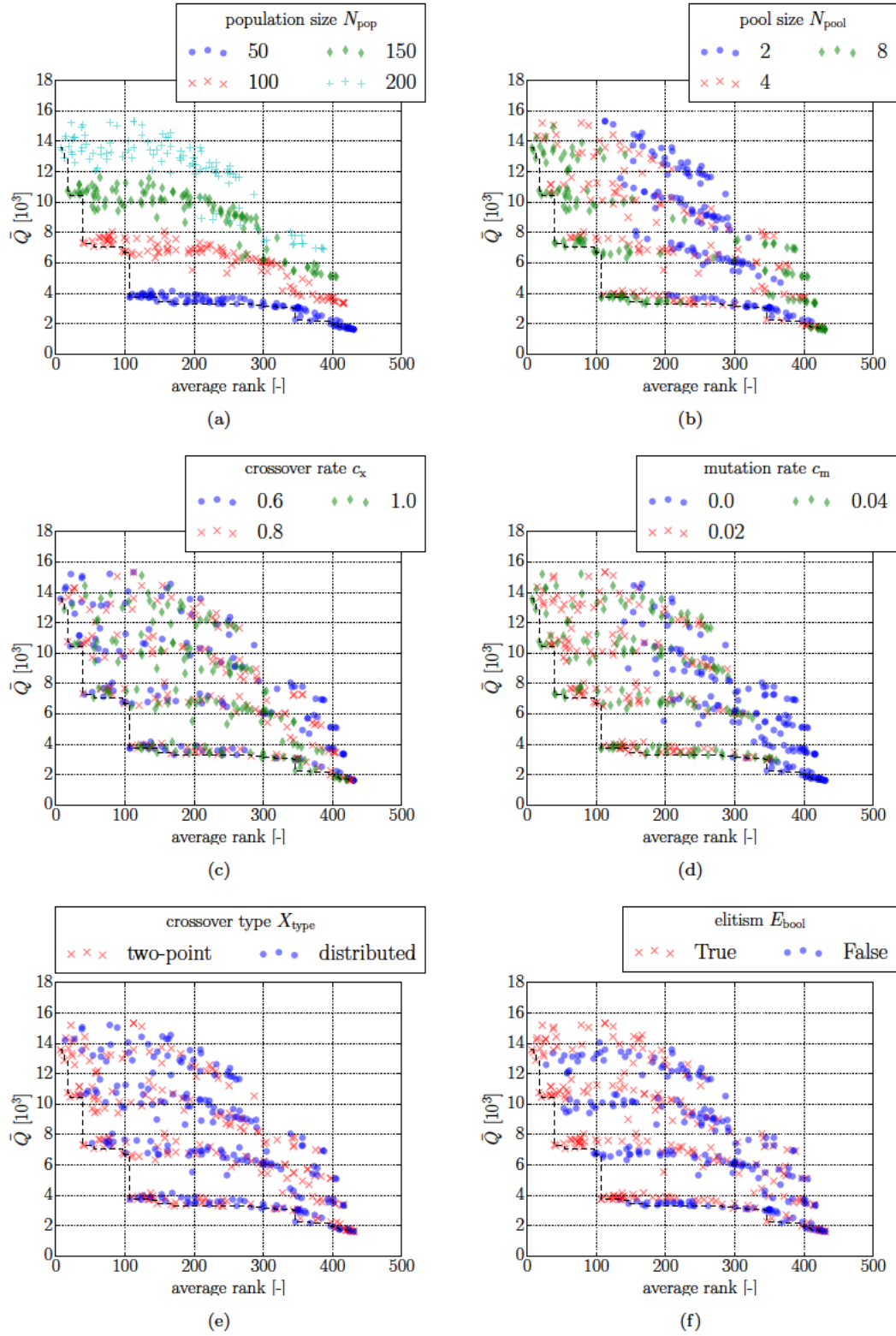


Figure 4.5: Results of the statistical experiment using the average rank. The parameter combinations are ranked based on their average minimized objective function value. The dashed line represents the Pareto front between minimum average number of function evaluations $\bar{Q}$ and minimum average rank.

#	N_{pop}	N_{pool}	c_x	c_m	X_{type}	E_{bool}	ranks (in test cases)					average rank	p_c^i [-]	$\bar{Q}$ [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

ⁱ Test Cases 1-4

Table 4.4: The parameter combinations lying on the Pareto front in Figure 4.5. The combinations are listed in increasing order of the average rank. An extended version of the table, including all tested parameter combinations, is openly available at <http://dx.doi.org/10.5258/SOTON/D0431>.

(a)		Test Case				
	ρ_s	1	2	3	4	5
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				
	ρ_s	1	2	3	4	5
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

Table 4.5: Pairwise Spearman’s rank correlation coefficients ρ_s between the test cases. The ranks are ordered based on the average minimized objective function value (a) and average number of objective function evaluations (b).

Tables 4.5(a) and 4.5(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 ($\rho_s = 1$). 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 $N_P = \{2 \dots 6\}$ on Test Case 1. As the optimization runs were repeated 70 times with $N_P = 4$ 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 Tables 4.6(a) and 4.6(b), using the same ranking measures as in Tables 4.5(a) and 4.5(b), respectively. The correlation coefficients, varying from 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 chapter is to offer practitioners of GA-driven L-Systems-based topology search advice on optimization setup, firmly grounded in empirical observations based

(a)		N_P				
	ρ_s	2	3	4	5	6
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				
	ρ_s	2	3	4	5	6
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

Table 4.6: Pairwise Spearman’s rank correlation coefficients ρ_s between a range rewriting rules $N_P = \{2 \dots 6\}$ on Test Case 1. The ranks are ordered in Subfigures a and b using the same measures as in Tables 4.5(a) and 4.5(b), respectively.

on a set of test problems. Later, in Chapters 5 and 7, we tackle structural geometry optimization problems 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.

4.6 Conclusions

The main objective of this chapter 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 that 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_s 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_s 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 chapter are also applicable to studies with a number of rewriting rules different to what we use here.

Later in this work, we will deploy the L-systems-based methods, with these control parameter guidelines, to engineering design problems. In the next section, we propose two other improvements to the map L-systems-based method.

Chapter 5

Map L-systems-based method: two proposed improvements

We propose two improvements to the map L-systems-based optimization method, which we refer to as the *linearization of the age variable* and the *component hierarchy*. The former is applicable to any L-systems-based geometry description, and the latter is applicable to the topology optimization of aircraft wing internal structure. As far as we are able to ascertain, neither of these variations is reported in the literature.

The proposed improvements are described in detail in Sections 5.1 and 5.2, respectively. We evaluate the performance of the linearized age variable by conducting another statistical experiment (Section 5.1.2), using the test cases defined in the previous chapter. Finally, in Section 5.3, we apply the proposed improvements to the design task of the small Unmanned Aerial Vehicle (sUAV) wing, presented in Section 3.7.

5.1 Linearization of the age variable

As we already mentioned in Section 2.7.4, the number of cell divisions in the map L-systems-based method increases quadratically with every developmental stage, if a matching marker pair is found inside its cells. The ordinal of the developmental stage, i.e. the age n , is typically used as a design variable when the method is used as a geometry description in topology optimization. Consequently, a small change in the age variable causes a major change in the geometry, which is not beneficial for the evolvability of the optimization method. In addition, the design variable is often encoded on a single element in a GA (e.g. the encoding by [Pedro and Kobayashi \(2011\)](#)). Let us refer to this variable as the *baseline age*. We propose an alternative age variable, referred to as the *linear age* n_{lin} , to replace the baseline design variable. The linear age defines the

total number of cell divisions in the system. We encode the linear age variable in binary format on multiple elements.

To demonstrate the change in the parameterization, let us return to the example map L-system presented earlier in the literature review (Figure 2.9). Starting from the axiom and ending to the fourth developmental stage, this system has the following numbers of cells: 1, 2, 4, 8, 16. Now, using the linear age variable, the same development in the system is shown in Figure 5.1. In this case, a small change in the age variable results in a small change in the geometry. At the same time, we expanded the design space (in this case) from five to 16 design candidates. The two alternative design variables are plotted against the total number of cells in Figure 5.2.

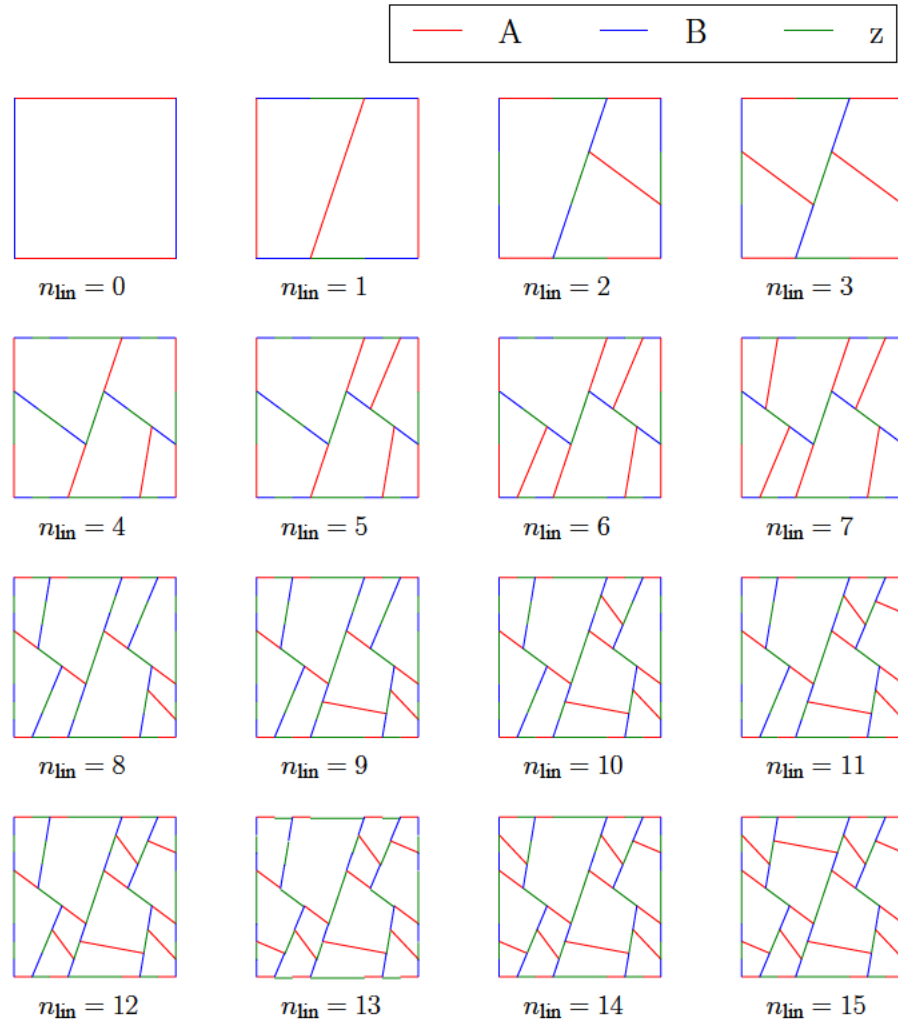


Figure 5.1: The development of the map L-system presented in Figure 2.9 using the number of cell divisions, n_{lin} , as the age variable.

If the defined number of cell divisions cannot be performed from the axiom and the rewriting rules, the individual is marked ‘unsuccessful’ and replaced by a new individual,

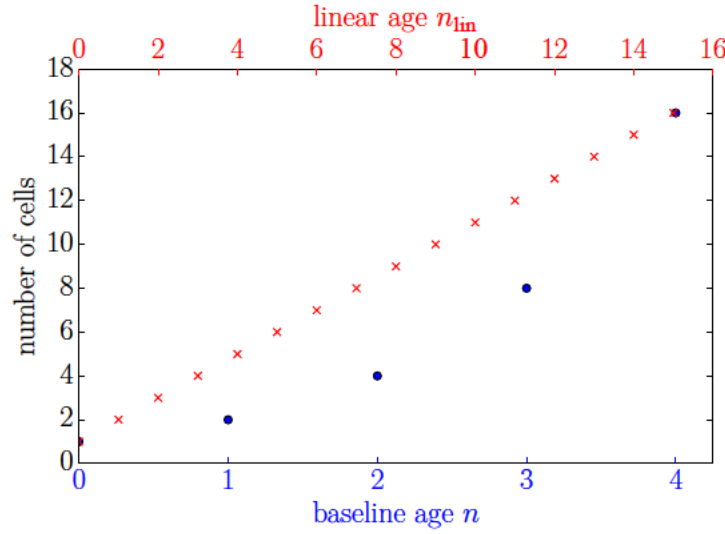


Figure 5.2: The number of cells in the map L-system described in Equation 2.24 as a function of the two alternative design variables: baseline age n (circles) and linear age n_{lin} (crosses). The baseline age is the ordinal of the developmental stage, whereas the linear age defines the number of cell divisions.

generated from the previous population via the genetic operators. The procedure does not involve any additional objective function evaluations.

Structural topology optimization often aims to determine the required number (and location) of a certain type of component. If the number is too low, the structure does not fulfill its strength requirements, and if it is too high, excess mass and complexity are added to the structure. The use of the linear age variable enables us to perform a local search, where the optimal number of edge divisions are searched for the corresponding axiom and rewriting rules. Therefore, we can evaluate the full potential of an individual, even if it has initially been assigned an unsuitable age variable.

We experiment the local search of the linear age n_{lin} by sampling first a test point $n_{lin,1}$ at a distance Δn_{lin} ¹ from the initial point $n_{lin,0}$, and then iteratively sampling points inside the interval, or if the fittest point is a boundary of the interval, outside the interval (the interval is extended from the side of the fittest point). All sampling points, after the first test point, are selected using a golden section between the fittest point and its neighboring point. To limit the additional computational cost, we perform the local search at every fifth generation (for all individuals), and limit the maximum number of local sampling points to four.

¹In this study we use a value of $\Delta n_{lin} = 4$

5.1.1 Demonstration on Test Case 1

Let us visualize the evolution process of a single optimization process, in which the age is parameterized using the number of cell divisions (i.e. linear age). We use again Test Case 1, devised in Section 4.2.1.

The evolution of the objective function, along with representative individuals, is shown in Figure 5.3. The global optimum is found at the 72nd generation with a map L-system defined as

$$\begin{aligned}
 \text{Axiom:} & \quad \omega_0 = CBCB \\
 \text{Rules:} & \quad \begin{aligned}
 P_1 : A &\rightarrow [+B][-\overleftarrow{C}]B \\
 P_2 : B &\rightarrow [+ \overleftarrow{C}][+D]C \\
 P_3 : C &\rightarrow [\overrightarrow{C}]D[-C][+\overleftarrow{B}]A \\
 P_4 : D &\rightarrow B[-D][+\overrightarrow{C}][-\overleftarrow{C}][+\overrightarrow{B}][-\overleftarrow{B}]
 \end{aligned} \\
 \text{Additional properties:} & \quad \begin{aligned}
 n_{\text{lin}} &= 15 \\
 f_a &= 0.3555
 \end{aligned}
 \end{aligned} \tag{5.1}$$

The obtained map L-system system is fundamentally different to the reference map L-system, presented in Equation 4.2. The design space was limited to an alphabet of four letters, and, as an outcome, all four letters are active in the system (the axiom introduced letters B and C , and the rewriting rule P_3 changes letter C into letters A and D , along with the defined markers). In addition, the linear age of $n_{\text{lin}} = 15$ corresponds in this case to a conventional age of $n = 6$. In contrast, the reference map L-system had only two active letters and the age of the system was $n = 4$. Therefore, the obtained system has found an alternative way to produce the same map as the reference system. The complete developmental process of the obtained system is presented in Figure 5.4.

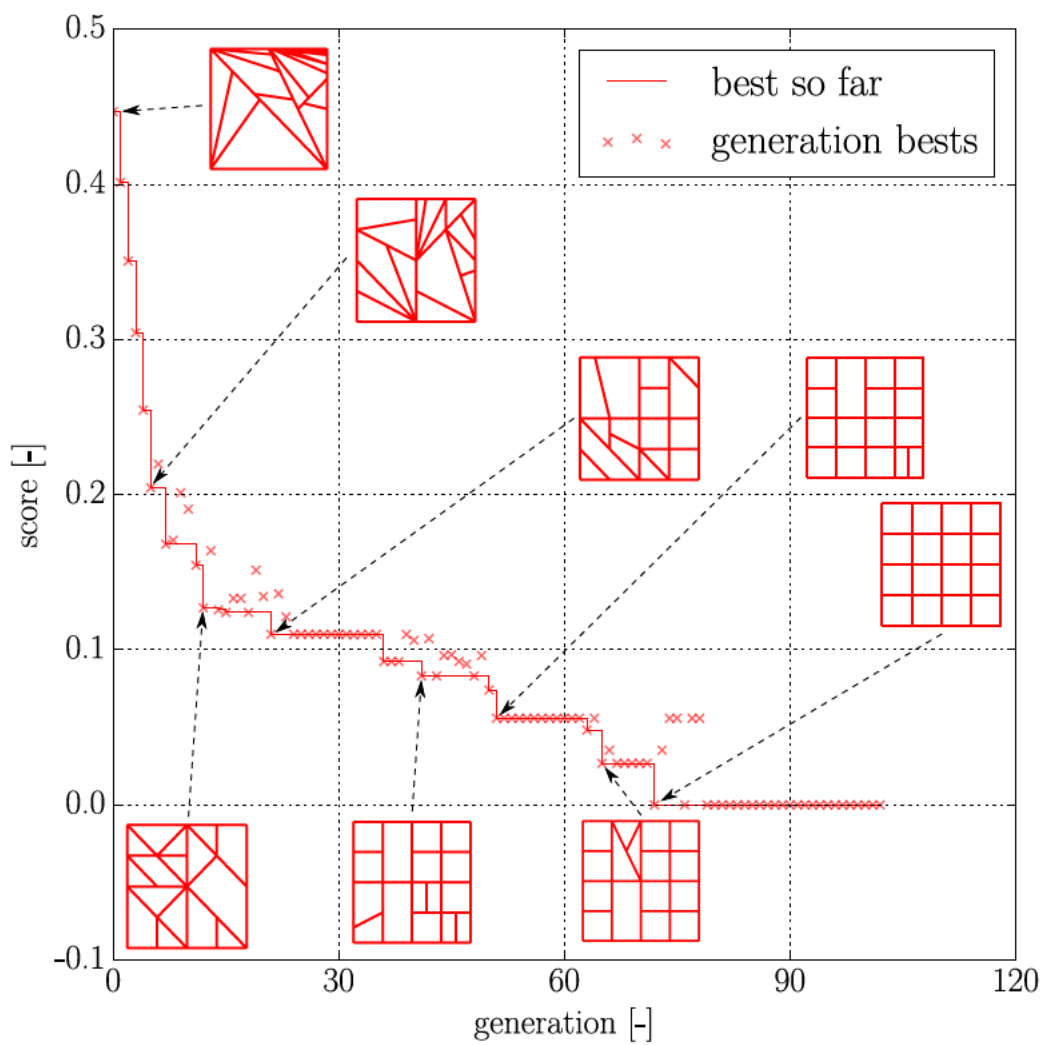


Figure 5.3: The best individuals of a GA optimization process, where the objective is to match the regular grid of 16 cells (Test Case 1). Representative individuals during the optimization process are shown.

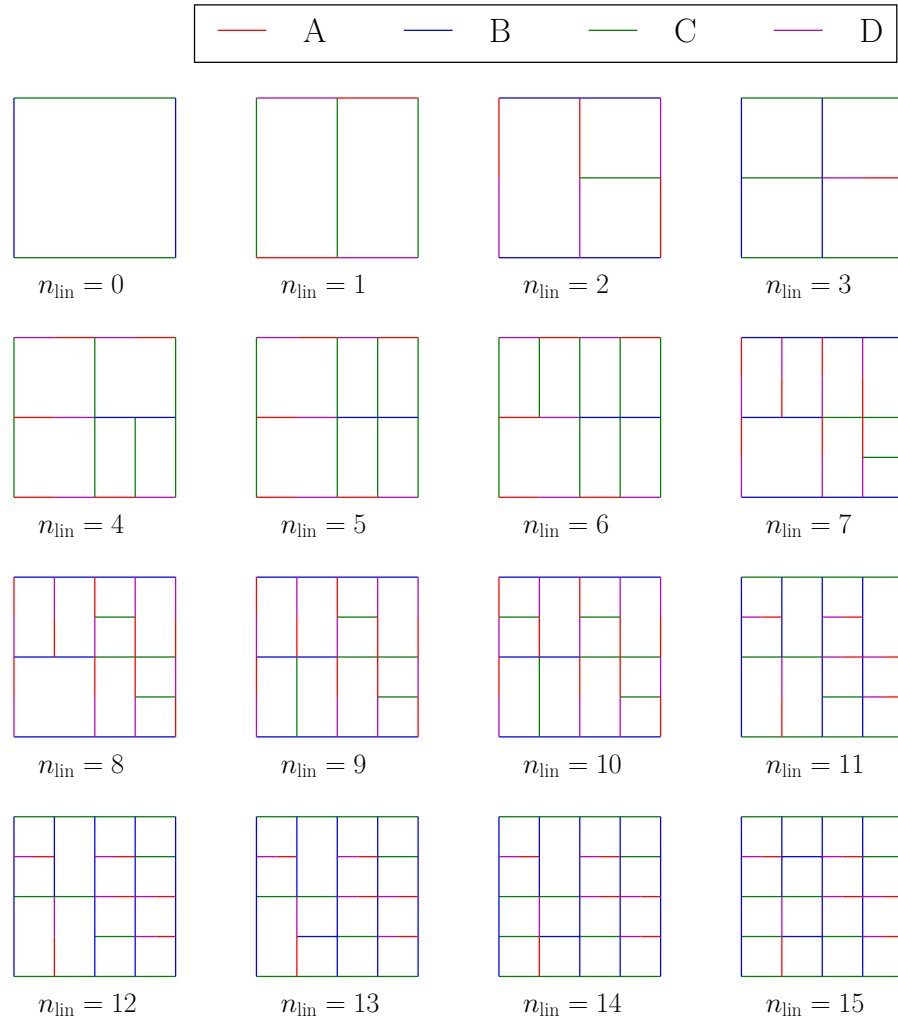


Figure 5.4: First 15 cell divisions of the map L-system described in equation 5.1. The cell divisions are performed at the following developmental stages: 1st (developmental stage): $n_{\text{lin}} = 1$, 2nd: $n_{\text{lin}} = 2$, 3rd: $n_{\text{lin}} = 3$, 4th: $n_{\text{lin}} = 4 \dots 6$, 5th: $n_{\text{lin}} = 7 \dots 10$, 6th: $n_{\text{lin}} = 11 \dots 15$

5.1.2 Statistical experiment

Let us next evaluate the performance of GAs, in which map L-systems are parameterized using the proposed linear age variable, against the corresponding GA, which use the baseline age variable. We evaluate the linear age parameterization with and without the previously presented local search algorithm (Section 5.1). Thus, the statistical experiment involves a total of three algorithms. We evaluate the performance of these algorithms on Test Cases 1-5, defined Sections 4.2.1 and 4.4.

Table 5.1 shows the ranges for the baseline and linear age variables, n and n_{lin} . The ranges are defined so that the maximum number of cells is equal, 64, in both design spaces. Other parameters defining the design spaces are the same as in Table 4.1.

variable	value
baseline age n	1 ... 6
linear age n_{lin}	0 ... 63

Table 5.1: The ranges of the alternative baseline and linear age variables.

In the previous chapter, we determined a Pareto set of recommended control parameter combinations (see Table 4.4). Here, we choose to use parameter combination #4. The reasoning is that, later in this chapter, we will apply the three algorithms to Application I, presented Section 3.7, for which we have already obtained results using the GA-based ground structure approach. The population size in the parameter combination #4 is equivalent, $N_{\text{pop}} = 150$, to that used in the ground structure approach. This choice will enable us to make a fair comparison between the two encoding methods at the end of this chapter.

We repeated each of the three algorithms 500 times on each of the five test cases. We evaluated the performance of the algorithms based on their completion rate p_c on finding the global optimum and their average number of objective function evaluations, $\bar{Q}$. The results of the statistical experiment are presented in Table 5.2. The confidence interval of 95% for the completion rate p_c is determined based on the Adjusted Wald method (Agresti and Coull, 1998). The corresponding confidence interval for the average number of objective function evaluations, $\bar{Q}$, is determined by multiplying its standard error by 1.96. The same results are plotted in Figure 5.5.

test case	age variable	local search	completion	best estimate	p_c [-]	95% confidence	mean	$\bar{Q}$ [10^3]	95% confidence
1	baseline	False	389/500	0.7769	[0.7395, 0.8123]			10.76	± 0.31
	linear	False	413/500	0.8247	[0.7902, 0.8568]			8.83	± 0.25
	linear	True	408/500	0.8147	[0.7796, 0.8476]			14.17	± 0.37
2	baseline	False	331/500	0.6614	[0.6194, 0.7021]			9.63	± 0.24
	linear	False	372/500	0.7430	[0.7039, 0.7803]			9.18	± 0.22
	linear	True	424/500	0.8466	[0.8138, 0.8769]			14.56	± 0.28
3	baseline	False	295/500	0.5896	[0.5464, 0.6323]			10.11	± 0.34
	linear	False	314/500	0.6275	[0.5848, 0.6693]			6.83	± 0.09
	linear	True	424/500	0.8466	[0.8138, 0.8769]			17.31	± 0.36
4	baseline	False	213/500	0.4263	[0.3834, 0.4697]			9.22	± 0.26
	linear	False	424/500	0.8466	[0.8138, 0.8769]			8.51	± 0.20
	linear	True	431/500	0.8606	[0.8289, 0.8896]			14.06	± 0.32
5	baseline	False	0/500	0.0020	[0.0000, 0.0065]			11.52	± 0.48
	linear	False	0/500	0.0020	[0.0000, 0.0065]			11.75	± 0.52
	linear	True	0/500	0.0020	[0.0000, 0.0065]			18.62	± 0.66

Table 5.2: Results of the statistical study on the baseline and linear age parameterizations. The linear age parameterization is used with and without the local search. Results for completion rate p_c and average number of objective function evaluations, $\bar{Q}$, are presented.

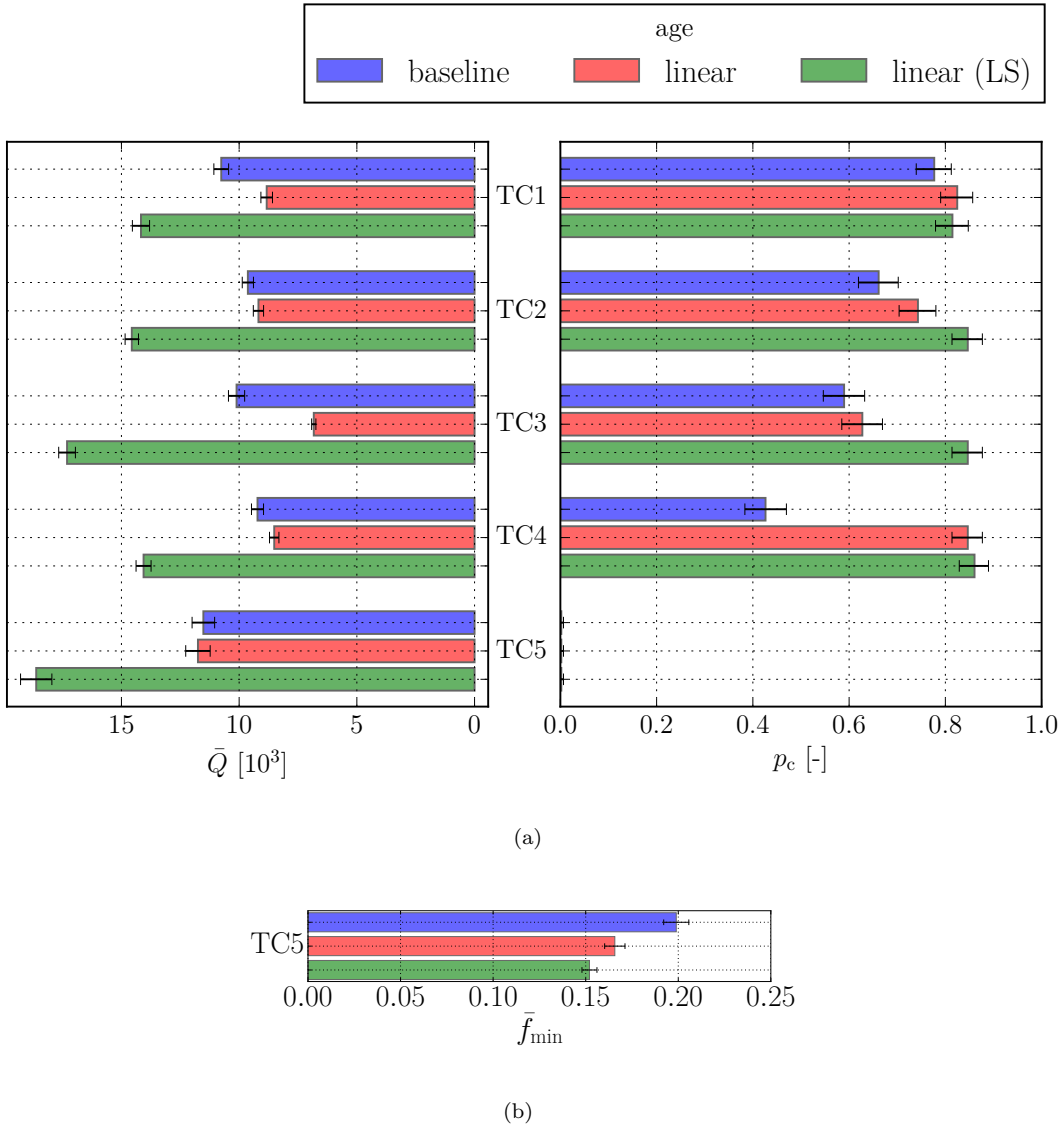


Figure 5.5: The completion rate p_c and average number of objective function evaluations, $\bar{Q}$, on the five test cases (TC) using algorithms with different age variables (a). The algorithm with the linear age variable is tested with and without the local search (LS). No global optimum was found for TC5; thus, we also report its average optimized objective function value $\bar{f}_{\min}$ (b).

The global optimum for Test Case 5 is not found with either the baseline or linear age parameterizations. Thus, for this test case, we report the average optimized objective function value $\bar{f}_{\min}$ of the algorithms in Figure 5.5(b). The confidence intervals of $\bar{f}_{\min}$ are reported as the standard error multiplied by 1.96.

We tested the statistical significance of these results, separately in each test case, by three pairwise tests of the two performance measures, i.e. the completion rate p_c ² and average number of objective function evaluations $\bar{Q}$ (yielding a total of 30 statistical

²In Test Case 5, we used the average optimized objective function value $\bar{f}_{\min}$, instead of the completion rate p_c .

tests). We used a significance level of $\alpha_s = 0.05$, and treated the inflated probability of false discoveries, due to performing multiple statistical tests in the same study, by the step-up false discovery rate (FDR) (Benjamini and Hochberg, 1995). With the FDR, the expected proportion of false statistically significant results among all statistically significant results is equivalent to the significance level α_s .

We obtained the following statistically significant results:

- The linear age parameterization (without the local search) yield higher completion rates, or lower optimized objective function value (in Test Case 5), than the baseline age parameterization in Test Cases 2, 4 and 5. In Test Cases 1 and 3, we are unable to make statistically significant conclusions.
- Considering the linear age parameterization, the use of the local search enhances completion rates, or optimized objective function value (in Test Case 5), even further in Test Cases 2, 3 and 5.
- However, the algorithm using local search (on the linear age parameterization) also requires the most objective function evaluations in all five test cases. This result was, in fact, expected as the local search, performed at every fifth generation, requires a significant number of additional objective function evaluations.
- Nevertheless, the linear age parameterization (without the local search) requires fewer objective function evaluations than the baseline age parameterization in Test Cases 1-4. In Test Case 5, the required numbers of objective function evaluations are similar.

We here present only the conclusions of the statistical tests; the reader interested in their details may consult Appendix A. Strictly speaking, all conclusions presented here are applicable to the tested five optimization problems only, though they may provide indications of what one may expect on similar problems.

To summarize, the linear age parameterization yields better final designs in three out of five test cases and requires fewer objective function evaluations in four out of five test cases than the baseline age parameterization. The local search algorithm further enhances the final designs with the linear age parameterization in most of the test cases, but results in a significantly increased number of required objective function evaluations.

Finally, the global optima in Test cases 1, 3 and 4 have fixed numbers of cells, which are 16, 64 and 12, respectively. One may argue that these fixed numbers are the reason for the better performance of linear age parameterization, and that ‘real-life’ topology optimization problems do not have these clear optimum numbers of cells. However, we observed that linear age parameterization also has a better performance than the baseline age parameterization on Test Cases 2 and 5, which do not have such fixed

optimum numbers of cells. We will apply the linear age parameterization, without the local search, to the design task of the sUAV wing later in Section 5.3.

5.2 Implementation of the component hierarchy

Aircraft wings with the effective use of material are often constructed using both full-depth structural members and stiffeners. In Section 3.3, we proposed a component hierarchy to be used as a part of the ground structure approach. Here, we propose the same component hierarchy to be used in the map L-systems-based optimization method. The purpose of the last component type of the hierarchy, (3) no structural member, is to add more design freedom and enable the formation of discontinuous structures. An edge, in a map L-system, that is assigned no structure serves as a ‘construction edge’ for other edges.

To further increase the design freedom, we introduce a new additional variable, called *maturity*. Maturity defines a threshold age, baseline or linear, prior to which constructed edges are converted into the hierarchy type (3) no structural member. We denote the baseline and linear maturity with symbols m and m_{lin} , respectively.

We include the variation of component types in the rewriting rules in a similar way as Pedro and Kobayashi (2011) included thickness transition in the rewriting rules (see the variable x_6 in Equation 2.28). In their definition, each token in the rewriting rules has an additional command that is either ‘decrease’, ‘retain’ or ‘increase’ the size of the offspring edge in comparison to the parent edge. With parent and offspring edges, they refer to the edge before and edges after an edge division, respectively. We make a small modification to the definition here to avoid excessively frequent changes in the component type. In our definition, the hierarchical type of an edge is retained in an edge division, but varied when introducing a new edge to the system, via connecting matching markers. Therefore, we refer to the parent edge as the edge, from which the new edge is initiated, and to the offspring edge as the edge that results from a cell division.

Pedro and Kobayashi (2011) did not encode initial edge thicknesses, but scaled the final thickness distribution to satisfy the mass constraint. As our application is not mass-constrained, and the sizes of the component types are fixed, we encode the initial component type into the numerical representation using elements $x_{p,0} \dots x_{p,N_a}$, such that

$$\mathbf{x} = \left[\underbrace{x_{a,1} \ x_{p,1} \ x_{a,2} \ x_{p,2} \ \dots \ x_{a,N_a} \ x_{p,N_a}}_{\text{Axiom } \omega_0} \underbrace{P_1 \ P_2 \ \dots \ P_{N_p}}_{\text{Rewriting rules } P_j} \underbrace{x_1 \ x_2 \ \dots \ x_{N_v}}_{\text{Additional variables}} \right]. \quad (5.2)$$

Each real number $x_{p,i}$ encodes the initial component type of the i th letter of the axiom ω_0 . Let us consider a component hierarchy of three types as an example; a real number

$x_{p,i}$ is assigned the following segments: $1 \equiv [0, \frac{1}{3}]$, $2 \equiv [\frac{1}{3}, \frac{2}{3}]$, $3 \equiv [\frac{2}{3}, 1]$. The encoding of other real numbers is the same as before.

Figure 5.6 presents an example of varying component types in a map L-system and the corresponding wing internal structure. These component types are drawn from the extended component hierarchy, which we defined in Equation 3.4.

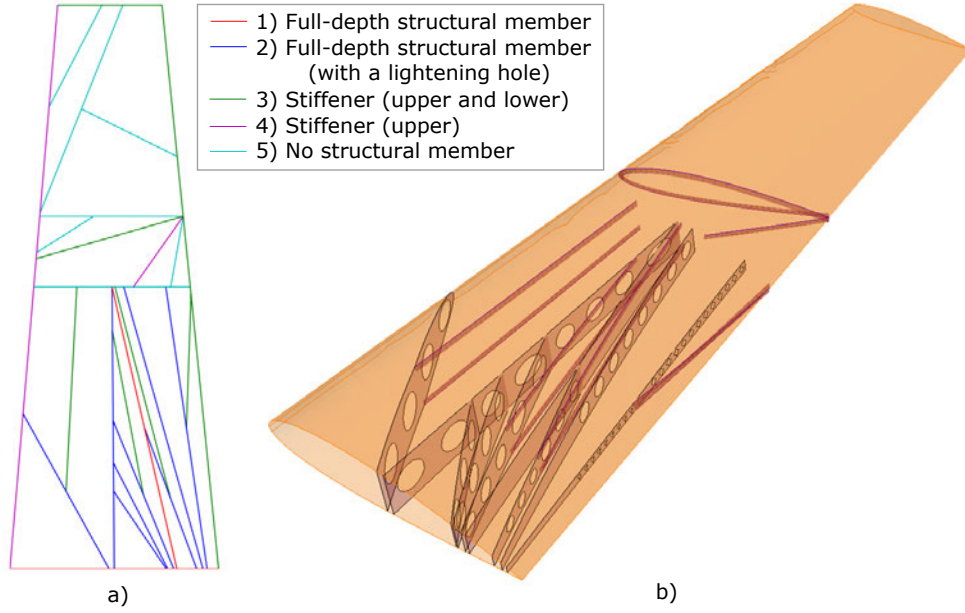


Figure 5.6: An example of mapping a map L-system with component hierarchy into a wing internal structure.

5.3 Application I continued: Topology optimization of a sUAV wing

In this section, we apply the map L-systems-based topology optimization, with the proposed improvements, to the design of the sUAV wing internal structure. Earlier in Section 3.7, we provided a detailed description of the design task and described results obtained using manual design methods and the ground structure approach.

The optimization procedure we use here is mainly the same as that defined in Figure 3.1 for the ground structure approach. The difference is that the ground structure approach operates on a predefined set of candidate structural members, which geometries were also generated (see ‘ground structure generator’ in Figure 3.1) prior to the iterative step of the optimization procedure (see ‘structural optimizer’ in Figure 3.1), but the map L-systems-based method does not recognize such candidate structural members. In the map L-systems-based method, geometrical definitions of structural members are only available after decoding the genotype. Thus, we embed the corresponding module,

which generates³ the internal structure of the wing, into the objective function evaluation loop before ‘FE analysis’⁴.

In Section 5.1.2, we discovered that, although providing slightly enhanced completion rates, the use of the local search significantly increases the number of required objective function evaluations. In the current application, the computational cost of an objective function evaluation is significantly higher than in the test cases. Therefore, we here apply the linear age parameterization only without the local search.

5.3.1 Defining the design space

We define the map L-system design space using the same main parameters as earlier with the low-cost test cases (Table 4.1) and the same component hierarchy of three options as in Section 3.7. However, we make the following minor changes to the design space.

First, we introduce two new additional variables: the minimum edge fraction of the minimum and maximum edge length in an offspring cell, f_e , and the minimum angle between two adjacent edges in an offspring cell, α_e (Table 5.3). These additional (fixed) variables prevent the formation of very small structural members and their alignments in small angles with respect to each other during the developmental process of phenotypes.

parameter	value
minimum edge fraction f_e	0.01
minimum edge angle α_e	5°
baseline age n (alternative)	1 ... 6
linear age n_{lin} (alternative)	1 ... 32
baseline maturity m (alternative)	1 ... 3
linear maturity m_{lin} (alternative)	1 ... 8

Table 5.3: Design space parameters of the sUAV wing topology optimization problem.

Second, we use both age and maturity (see Section 5.2) variables, the ranges of which are given in Table 5.3. We have adjusted the range of the linear age variable from 0 ... 63 to 1 ... 32, due to the following reasons. 1) Based on earlier results, we know a priori that the wing structure without any internal structure ($n_{\text{lin}} = 0$) does not withstand the design load without buckling. 2) The optimized internal structures, obtained using the ground structure approach, consist of significantly fewer than 63 structural members, which was the earlier upper bound.

³Using the Python scripting interface of Rhinoceros.

⁴We perform the FE analyses using the representative element mesh size of 5 mm, which was verified to be adequate in 3.7.4.1.

5.3.2 Results and discussion

We repeated the optimization process 10 times with algorithms using the linear and baseline age parameterizations, n and n_{lin} . Table 5.4 lists the average and standard deviation of both the optimized objective function values, i.e. internal structure mass m_{IS} , and the required objective function evaluations, $\bar{Q}$. The table also includes corresponding results of the manually generated stiffener design and those obtained using the ground structure approach.

method	graph	age variable	m_{IS} [g]	$\bar{Q}$ [10^3]
manual design	Figure 3.8(d)	-	7.67	-
GS1	Figure 3.19(m)	-	3.21 ± 0.14	13.47 ± 1.17
GS3 (Density 3)	Figure 3.21(g)	-	2.59	31.06
map L-system	Figure 5.8(a)	n	6.41 ± 1.17	12.56 ± 5.26
map L-system	Figure 5.8(b)	n_{lin}	5.92 ± 0.79	13.37 ± 2.91

Table 5.4: Comparison of the optimized internal structure mass, m_{IS} , and required objective function evaluations, $\bar{Q}$.

Let us first compare the results of the map L-systems-based method using the linear and baseline age parameterizations. On average, the linear age parameterization yields slightly lower optimized mass, but also requires slightly more objective function evaluations than the baseline age parameterization. However, these results are not statistically significant. Using the two-tailed non-parametric Mann-Whitney U-test, the p-values of these comparisons are 0.120 and 0.367, respectively. We have plotted the distributions of these two quantities in Figure 5.7, along with the corresponding results using the ground structure approach⁵.

The map L-systems-based method clearly yields worse designs than the ground structure approach in the current application. The internal structure of the designs obtained by the linear age parameterization is on average 84.4% heavier than of those obtained using GS1. In addition, the best design obtained by the map L-systems-based method has 79.1% heavier internal structure than the best design obtained by the ground structure approach (using GS3: Density 3). However, 19 out of 20 map L-systems-based optimization runs (with baseline and linear age parameterizations) still yield optimized designs with lower mass m_{IS} than that of the manually designed stiffener design.

Figures 5.8(a) and 5.8(b) present the lightest designs obtained by the map L-systems-based method with the baseline and linear age parameterizations, respectively. Moreover, Figure 5.9 visualizes the evolution process of the latter. These designs have some similarities to the designs obtained by the ground structure approach; the dominant

⁵We here plot the results obtained using GS1, since it is the only ground structure which we used to perform repeated optimization runs. These results are obtained using the best performing crossover operator, i.e. the distributed crossover (see Section 3.7.4.4).

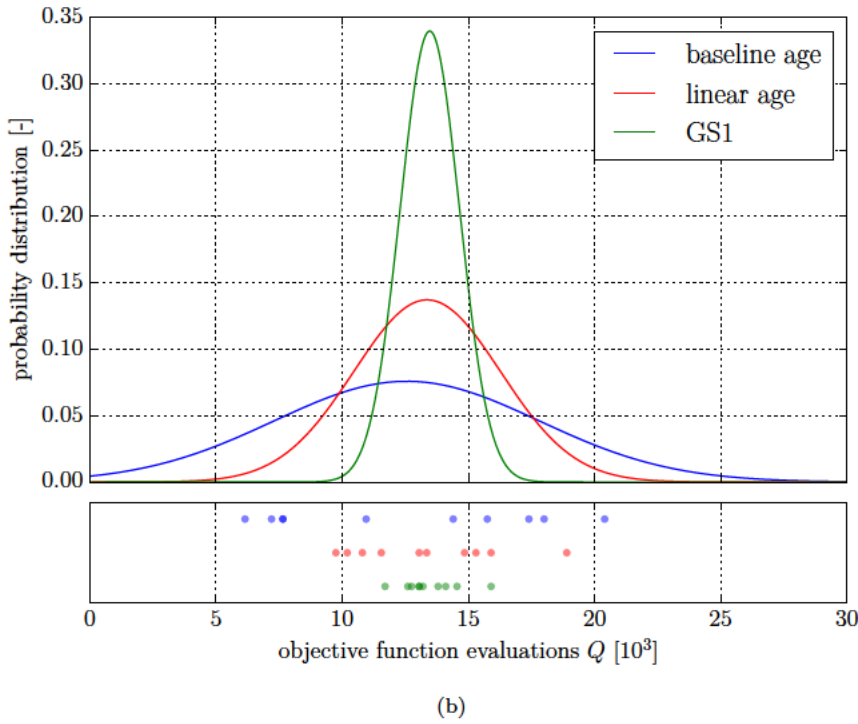
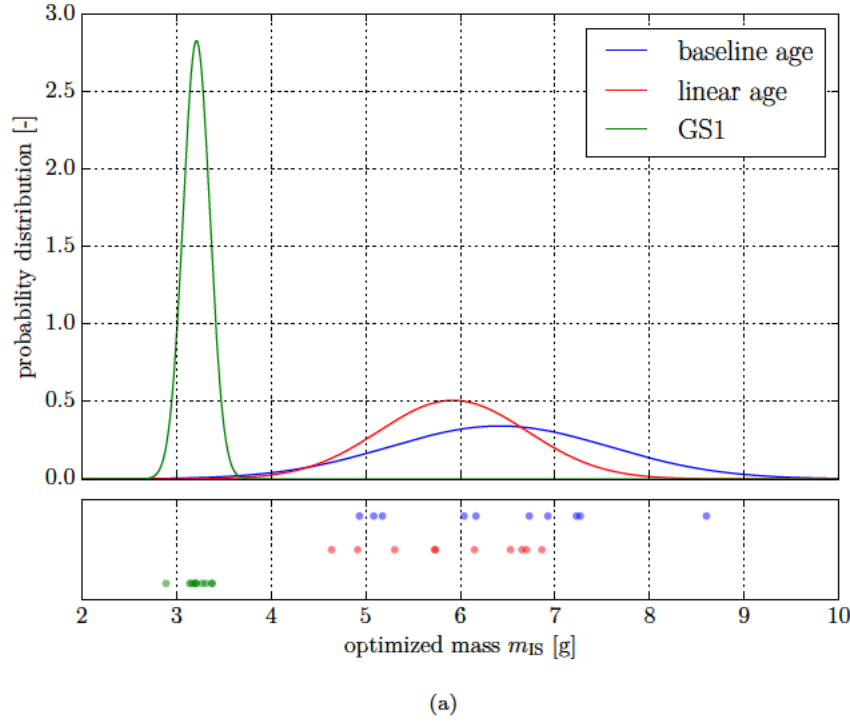


Figure 5.7: Distributions of optimized mass m_{IS} (a) and required objective function evaluations, Q (b), using the map L-systems-based method with baseline and linear age parameterizations. As a reference, we have included the corresponding results obtained using the ground structure approach with GS1.

structural member type is the stiffener on the upper skin, and the two skins are connected by one or two full-depth structural members. These structural members are located in the same region of the wing: in the vicinity of the root and trailing edge of the wing, which we found earlier to be the most prone to buckling. It is also noticeable that each of the three structural member types, drawn from the component hierarchy, is present in the designs.

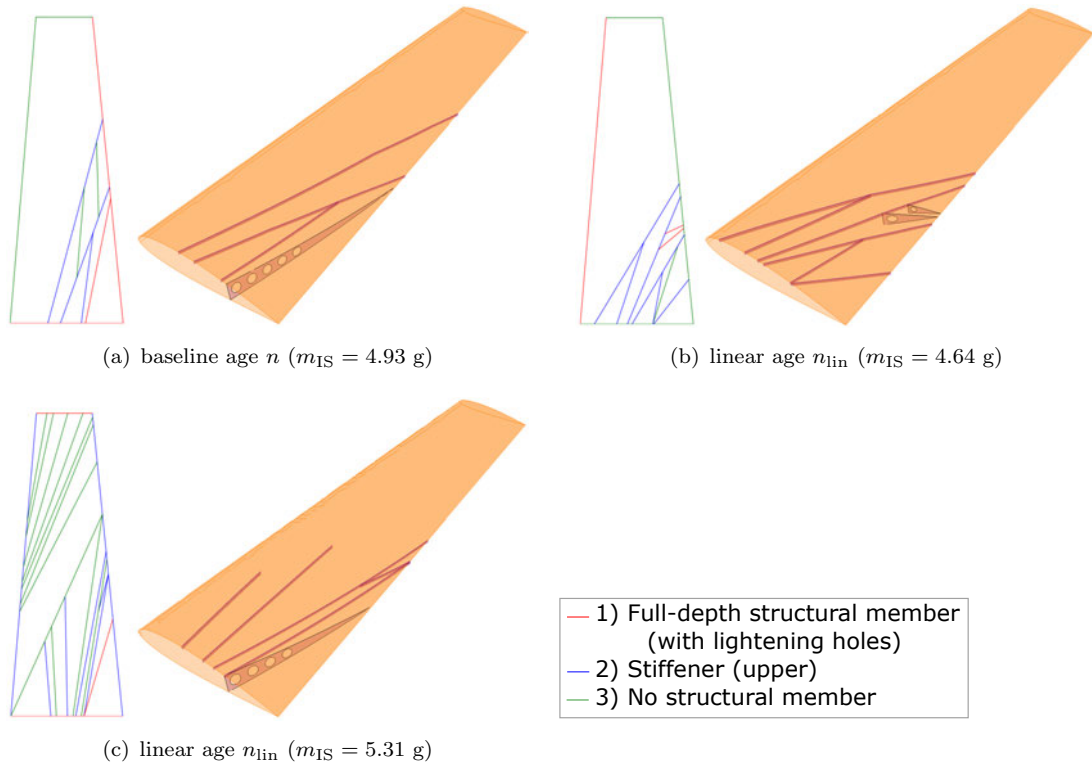


Figure 5.8: Optimized designs obtained using the map L-systems-based method.

However, these design do not have structural discontinuities, which we encountered in the designs obtained by the ground structure approach. Nevertheless, structural discontinuities are present in other (heavier) designs that we obtained using the map L-systems-based method. Figure 5.8(c) presents one of these designs, in which two stiffeners on the upper skin terminate at around one third of the semispan. These stiffeners are similar to those seen in the designs obtained by the ground structure approach.

Map L-systems naturally yield phenotypes which do not have structural discontinuities. Our aim here was to enable the formation of discontinuities by including the structural member type ‘no structural member’ in the member hierarchy and introducing the maturity (m or m_{lin}) as one of the additional variables. We believe that the main reason why the map L-systems-based method yields worse designs than the ground structure approach is that, despite these efforts, we were not able to define a parameterization that

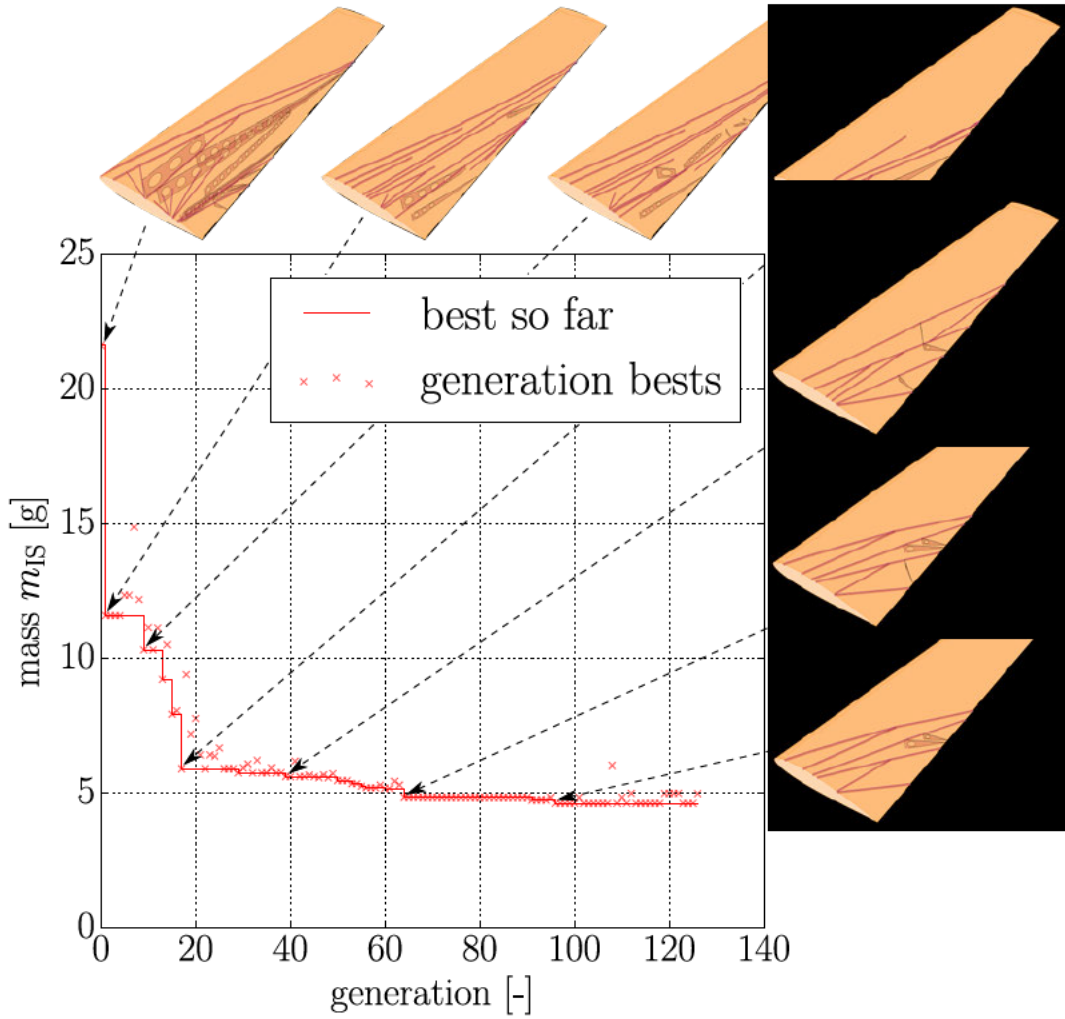


Figure 5.9: The evolution process of design shown in Figure 5.8(b).

is flexible enough to yield localized and discontinuous internal structure in the extent of the lightest known design (Figure 3.21(g)).

Finally, we want to emphasize that the presented application is a special case of an aircraft wing. Considering the relatively large minimum wall thickness constraint due to the manufacturing via 3D printing, and the resulting inactive von Mises stress constraint, the only purpose of the internal structure is to prevent the wing from buckling under the defined design load. Under these circumstances, the structural discontinuity seems to be a favorable feature of the internal structure.

However, in design optimization of larger aircraft wings, as well as other type of engineering structures, the stress constraint is typically active and, thus, structural continuity is highly desired. In this case, the map L-systems-based method has the potential to be useful as designs with undesired structural discontinuities can be eliminated from the

design space. We will return using the map L-systems-based method again in Chapter 7.

5.4 Conclusions

In this chapter, we presented two variations to the map L-systems-based topology optimization method. The first is to linearize its age variable. We conducted a statistical experiment on Test Cases 1-5, defined in the previous chapter, to benchmark the performance of the linearized age parameterization against the baseline age parameterization. The results show statistical evidence to support the hypothesis that, in comparison to the baseline age parameterization, the use of the linear age parameterization yields better designs, in three out of five test cases, and reduces the number of required objective function evaluations, in four out of five test cases. We ran experiments of these two age parameterizations also on the sUAV wing application, but, due to the small sample size, were not able to obtain statistically significant results.

In addition, we evaluated the performance of the local search algorithm on the linearized age variable. We show statistical evidence that the algorithm enhances the fitness of the optimized designs even further. However, the use of the algorithm increases the number of required objective function evaluations by around 60% to 150%, depending on the test case.

The second variation to the method is to include a component hierarchy, from which types of structural components (i.e. a full-depth structural member or a stiffener) may or may not appear in the structure. We embedded the rules to vary these types in the rewriting rules of the map L-systems, using a similar approach to that of [Pedro and Kobayashi \(2011\)](#) who varied the thickness distribution of their cantilever structure. We demonstrate the use of the component hierarchy on the design task of the sUAV wing internal structure.

Finally, we benchmarked these results against the ground structure approach. Using representative parameter choices with both methods, the map L-systems-based method yielded, on average of 10 optimization runs, 84.4% heavier designs than the ground structure approach.

Chapter 6

Application II: Heat conductor

So far in this work, we have represented designs only by one type of interpretation formalism of L-systems, map L-systems. In this chapter¹, we apply another type of interpretation formalism, the turtle interpretation (Section 2.7.1), to represent the distribution of high conductive material inside an electrical device.

Electronic devices are packed in increasingly compact spaces, which increases the heat density generated by their components. To prevent overheating, their architecture must be designed with an effective cooling system. The first task of the cooling system is to conduct the heat from the electronic components to a heat sink, using highly conductive material, e.g. copper or aluminum. The availability of conductive material is limited by space constraints and because the manufacturers always wish to reduce the cost of such components. Consequently, properly distributing the high conductivity material through a finite volume becomes an important topology optimization problem.

In Section 2.8.2, we found that the majority of studies in the literature on such optimization problems report optimized designs resembling of a branching tree-structure. Moreover, one of the authors, Dede (2009), described their designs to have self-similar features. In Section 2.9, we concluded that the turtle interpretation of L-systems naturally yields such structures, but they have not directly been applied to the optimization problem. Therefore, the objective of this chapter is to implement L-systems and its turtle interpretation as a parameterization method for evolutionary topology optimization, and apply them to the conductive heat transfer problem defined by Bejan (1997).

As far as we are able to ascertain, the only similar approach in the literature is that of Kobayashi (2010). However, he defines the optimization problem to represent design optimization of an artificial cordate leaf, instead of the cooling of an electrical device as originally defined by Bejan (1997). Therefore, his design domain has the shape of a leaf and he uses the pressure drop as one of the objectives. The pressure drop is

¹The work of this chapter has been conducted in collaboration with Dr. Gilles Marck. The distinction of his and the author's contributions to the work is described in Appendix C.

meaningless in the context of the design problem studied here. Due to these aspects, his results cannot be benchmarked against other studies on the original design problem. We benchmark our single-objective optimization results against relevant studies in the literature, and demonstrate the use of the methodology in multi-objective optimization with relevant temperature and manufacturing related objectives.

One of the aims in this work is to identify engineering design problems to which generative encodings are particularly suitable, or unsuitable, in comparison to direct encoding methods. Therefore, we also apply the ground structure approach to the optimization problem.

6.1 Optimization problem

The optimization problem defined by [Bejan \(1997\)](#) represents an electrical device that is to be cooled by distributing limited amount of high conductive material inside its package. The purpose of this material is to conduct the heat to a heat sink, located at the boundary of the package.

Let us consider a two-dimensional square-shape design domain Ω , with a side length l . The domain consists of two subdomains Ω_p and Ω_0 , such that $\Omega_p \cup \Omega_0 = \Omega$ and $\Omega_p \cap \Omega_0 = \emptyset$ (Figure 6.1). Subdomains Ω_p and Ω_0 denote high and low conductive materials with thermal conductivities k_p and k_0 , respectively. The latter represents the area of the device that is filled with electrical components, and thus is defined to have heat-generation rate² q . The design domain is bound by Dirichlet and Neumann boundary conditions, Γ_D and Γ_N (Figure 6.1). The Dirichlet boundary condition (heat sink) is located in the middle of the left-hand side boundary and has a width of d , whereas the remaining boundary conditions are adiabatic (Neumann). Thus, the governing equations for steady-state conductive heat transfer in the domain are

$$\begin{cases} \nabla \cdot (k \nabla T) + q = 0 & \text{on } \Omega \\ (k \nabla T) \cdot \mathbf{n} = 0 & \text{on } \Gamma_N \\ T = 0 & \text{on } \Gamma_D, \end{cases} \quad (6.1)$$

where $\mathbf{n}$ is the outward normal vector of the boundary, k is the local thermal conductivity, that is k_p or k_0 corresponding to subdomains Ω_p and Ω_0 and q is the local heat generation rate, that is q_0 in domain Ω_0 and 0 elsewhere.

²In fact, [Bejan \(1997\)](#) defined the entire domain to have an evenly distribute heat-generation rate. We have eliminated the heat generation from subdomain Ω_p as the same choice is made by [Boichot and Fan \(2016\)](#), whose results we use as a benchmark. However, the definition by [Bejan \(1997\)](#) is a closer representation of a real chip, which high conductive layer is on top/bottom.

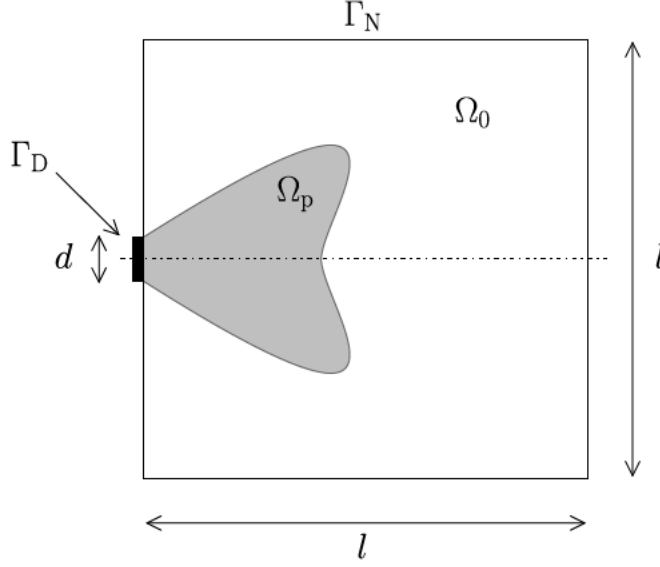


Figure 6.1: The optimization problem of optimally distributing a limited amount of high conductive material (domain Ω_p), to minimize the average, or maximum, temperature in domain $\Omega_p \cup \Omega_0$.

In view of optimal design studies, it is convenient to introduce the characteristic function of Ω_p , $\chi_{\Omega_p} : \Omega \rightarrow \{0, 1\}$, defined by

$$\chi_{\Omega_p}(x) = \begin{cases} 1 & \text{if } x \in \Omega_p \\ 0 & \text{if } x \in \Omega_0. \end{cases} \quad (6.2)$$

This allows us to define the scalar variables as a function of χ_{Ω_p} , such that the thermal conductivity is

$$k(\chi_{\Omega_p}) = k_0 + (k_p - k_0)\chi_{\Omega_p} \quad (6.3)$$

and the heat generation rate is

$$q(\chi_{\Omega_p}) = q_0(1 - \chi_{\Omega_p}). \quad (6.4)$$

Then, the optimization problem becomes

$$\begin{aligned} & \min_{\chi_{\Omega_p} \in \mathcal{D}_\phi} f(\chi_{\Omega_p}) \\ & \text{subject to} \quad \nabla \cdot (k(\chi_{\Omega_p}) \nabla T) + q(\chi_{\Omega_p}) = 0 \quad \text{on } \Omega, \\ & \quad (k(\chi_{\Omega_p}) \nabla T) \cdot \mathbf{n} = 0 \quad \text{on } \Gamma_N, \\ & \quad T = 0 \quad \text{on } \Gamma_D, \\ & \text{where} \quad \mathcal{D}_\phi = \{\chi_{\Omega_p}, |\Omega_p| \leq \phi|\Omega|\} \end{aligned} \quad (6.5)$$

where $\phi \in [0, 1]$ is the volume constraint, restricting the area covered by the domain Ω_p to a fraction of the whole finite-size volume Ω .

Unless mentioned otherwise, the objective function $f(\chi_{\Omega_p})$ is either the average temperature $\bar{T}$ or the maximum temperature $T_{\max}$, defined as

$$\begin{cases} \bar{T} = \frac{1}{|\Omega|} \int_{\Omega} T d\Omega \\ T_{\max} = \arg \max_{x \in \Omega} T(x). \end{cases} \quad (6.6)$$

6.2 Methods

In this section, we describe in detail, first, the means of representing design candidates via the ground structure approach and parametric L-systems and, second, the procedure of evaluating the temperature fields of these designs. Figure 6.2 illustrates the optimization procedure, which objective function evaluation consists of the following three steps.

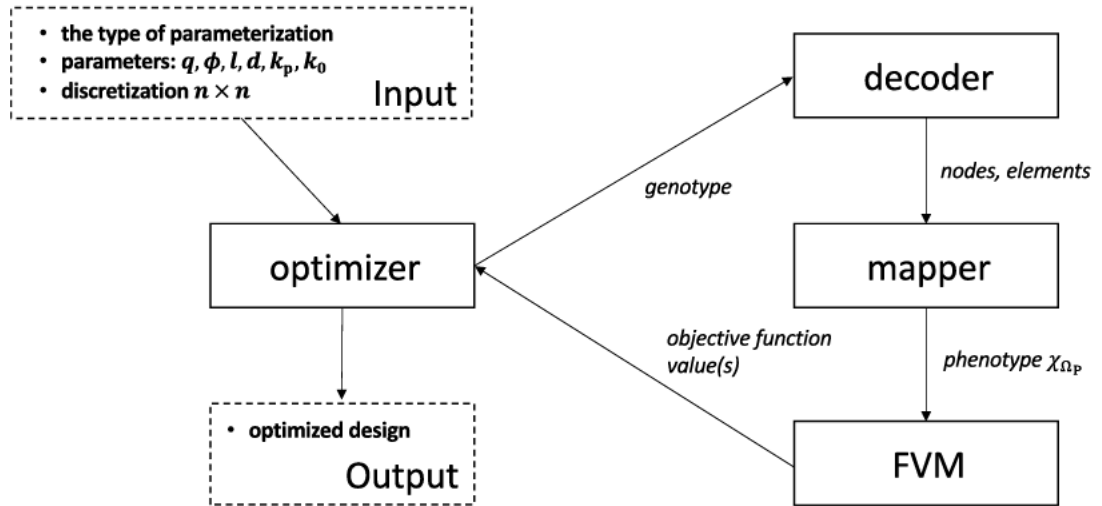


Figure 6.2: The procedure of the optimization process.

First, the decoder interprets the genotype of the design candidate, which is a vector of real numbers $\mathbf{x}$, with $x_i \in [0, 1]$, into a format of nodes and elements. This format is well-suited to describe geometries having a bifurcating tree structure (see an illustration in Figure 6.3), which we found the reported optimized designs in the literature to resemble. In addition to the two nodes, we also assign a non-dimensional width, or widths³, to these elements. Two alternative decoders, based on the ground structure approach and the L-systems-based method, are described in Sections 6.2.1 and 6.2.2, respectively.

Second, as the optimization problem is volume-constrained, the mapper (Section 6.2.3) first scales the non-dimensional element width(s) so that the total volume of high conductive material elements satisfies the volume fraction ϕ . The scaling is performed using

³In the L-systems-based method, we assign non-dimensional widths to the start and end of an element. The non-dimensional width of the element varies linearly between these two points.

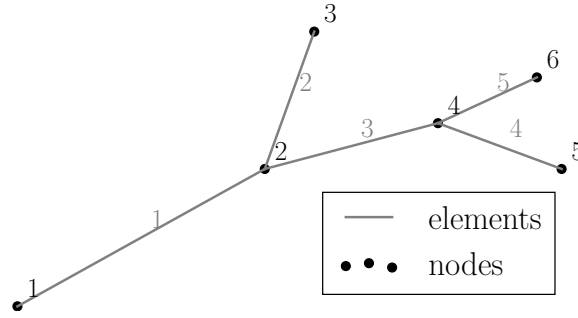


Figure 6.3: An example of a bifurcating tree-structure, represented using nodes 1-6 (in *black*) and elements 1-5 (in *grey*).

a scaling factor c_{sca} , which retains the relative differences of the non-dimensional widths of the elements. The mapper then projects the elements to the design domain grid, yielding the characteristic function χ_{Ω_p} , which defines the distribution of high conductive material in the phenotype.

Third, the temperature field of the design is solved using the Finite Volume Method (FVM) (Section 6.2.4) and, finally, the objective function value(s), i.e. the average and/or maximum temperature of the domain Ω , is returned to the optimizer.

As the optimizer, we again apply genetic algorithms (GAs). However, this time we implement GAs using the open-source Python package DEAP (Distributed Evolutionary Algorithm for Python) (Fortin et al., 2012). The reason is that later in this chapter we study multi-objective optimization problems where we evolve the L-systems encoding by the NSGA-II (Section 2.3.4). Contrary to Pyevolve, which we have used earlier in this work, DEAP contains an implementation of NSGA-II.

As the optimization problem is symmetric, we assume the optimal distribution of high conductive material also to be symmetric, and therefore analyze only the upper part of the design domain. The same assumption has been made in most of the studies in the literature. Xu et al. (2007) conducted their optimization studies on the whole design domain, but their optimized designs are also nearly symmetric. When analyzing the half-domain, we define its lower boundary (the dash-dotted line in Figure 6.1) to be adiabatic, and therefore a part of the Neumann boundary Γ_N .

6.2.1 Ground structure approach

The simplest way of applying the direct encoding to the optimization problem is to explicitly assign a binary design variable to each of the discrete material elements. As we found in the literature review (Section 2.8.2), this approach was used by Boichot and Fan (2016). On the other hand, Xu et al. (2007) encoded an individual design as a list of high conductive material locations. However, while Xu et al. (2007) do not report the

computational cost of their algorithm, [Boichot and Fan \(2016\)](#) report their algorithm, containing 5000 design variables⁴, to require more than five million objective function evaluations for full convergence. In addition, the method lacks of an elegant way of preventing the individuals from violating the volume fraction constraint⁵ ϕ .

To avoid the high computational cost, we adopt an alternative direct encoding approach. We define a ground structure consisting of candidate path sections for the optimized structure (Figure 6.4). These sections are defined using the aforementioned format of nodes and elements (and element widths). The approach not only significantly reduces the number of design variables, but also offers a convenient way of scaling widths of the ground structure members so that the total volume of high conductive material satisfies the volume fraction constraint ϕ .

Here we use the same layouts of ground structures as in Application I (Section 3.7): the quadrilateral with diagonals and hexagonal ground structures. Also here, the structural members may have different types/widths.

Later in this chapter, we will vary the density of these ground structures. Let us now define a common measure for the density of the two ground structure types as $m \times n$, where m and n are numbers of segments the ground structure members divide the west and north boundaries into, respectively. Thus, the density of both ground structures in Figure 6.4 is 8×4 .

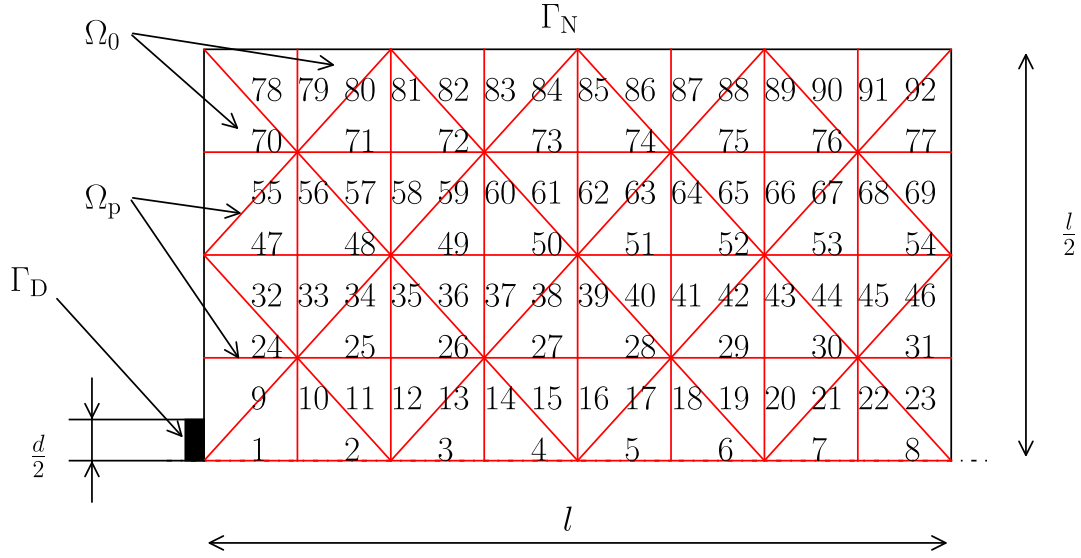
The existence of structural members, and their non-dimensional widths, are encoded into a genotype $\mathbf{x}$, having $N_{\text{gs}} + 1$ elements, where N_{gs} is the number of ground structure members. To decode the content of the genotype, its first N_{gs} elements are interpreted as follows:

- if $x_i \in [0, \frac{1}{N_w+1}]$, structural member i exists, and its non-dimensional width is $w_{\text{frac}}^0 = 1$
- else, if $x_i \in [\frac{1}{N_w+1}, \frac{2}{N_w+1}]$, structural member i exists, and its non-dimensional width is w_{frac}^1
- else, if $x_i \in [\frac{2}{N_w+1}, \frac{3}{N_w+1}]$, structural member i exists, and its non-dimensional width is w_{frac}^2
- ...
- else, if $x_i \in [\frac{N_w}{N_w+1}, 1]$, structural member e_i does not exist,

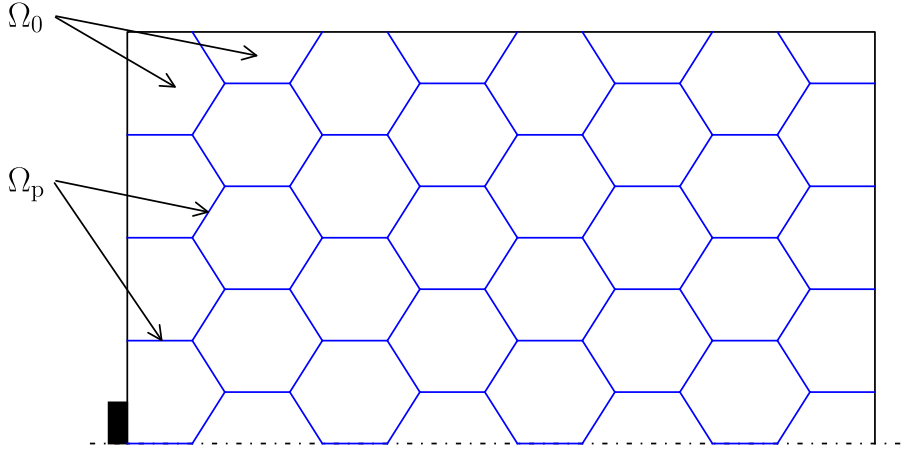
where i is the ordinal of the element in the genotype, and N_w is the number of different non-dimensional widths. Figure 6.4(a) illustrates the order in which the structural

⁴They used the half-domain, and discretized it into 100×50 material elements.

⁵[Boichot and Fan \(2016\)](#) repaired infeasible individuals by randomly removing material elements, until the constraint was satisfied.



(a) Rectangular ground structure with diagonal members



(b) Hexagonal ground structure

Figure 6.4: Two types of ground structures providing candidate paths for the high conductive material in the half-domain (the upper half of Figure 6.1). The numbering in subfigure (a) indicates the order, in which the structural members are encoded into the genotype.

members are encoded in the case of 8×4 quadrilateral ground structure with diagonals. The variable w_{frac} is the fraction between two non-dimensional widths, and is encoded into the last element of the genotype, as $x_{N_{\text{gs}}+1} = w_{\text{frac}}$.

Finally, it is worth mentioning that this approach has some features of geometric encoding (Section 2.2.2), as the existence of material elements do not only depend on a single design variable, but also on the total number of existing ground structure members and their non-dimensional widths. However, as it is based on the ground structure approach, we regard it still as a direct encoding method.

6.2.2 L-systems-based method

We reviewed the turtle interpretation of L-systems in section 2.7.1, and further, of parametric L-systems in Section 2.7.1.1. The L-systems-based method, we use in this chapter, is based on the turtle interpretation of parametric L-systems.

When such grammatical models are evolved via a GA, a key implementation detail is how to encode them into a numerical format suitable for the algorithm. Here we use a modified version of the numerical representation defined by Kobayashi (2010), which he developed to represent the venation topology of an artificial cordate leaf.

We encode the axiom and rewriting rules, as well as some additional variables sequentially into a vector $\mathbf{x}$ of real numbers, with $x_i \in [0, 1] \forall i$, as

$$\mathbf{x} = \left[\underbrace{x_{a,1} \ x_{a,2} \ \dots \ x_{a,N_a}}_{\text{Axiom } \omega_0} \ \underbrace{\mathbf{y}_1 \ \mathbf{y}_2 \ \dots \ \mathbf{y}_{N_P}}_{\text{Rewriting rules } P_i} \ \underbrace{x_{c,1} \ x_{c,2} \ \dots \ x_{c,N_v}}_{\text{Additional variables}} \right]. \quad (6.7)$$

The axiom ω_0 consists of N_a letters, each of which are represented by a real number $x_{a,i}$. The interval $[0, 1]$ of the real number is divide into equally sized segments that represent the letters in the alphabet Σ . As an example, if the alphabet contains letters $\{A, B, C, D\}$, the encoding is the following:

$$\begin{aligned} \text{if } x_i \in [0, \frac{1}{4}] &\rightarrow A \\ \text{if } x_i \in [\frac{1}{4}, \frac{1}{2}] &\rightarrow B \\ \text{if } x_i \in [\frac{1}{2}, \frac{3}{4}] &\rightarrow C \\ \text{if } x_i \in [\frac{3}{4}, 1] &\rightarrow D \end{aligned} \quad (6.8)$$

Each letter σ_i in the alphabet Σ , containing a total of N_P letters, is assigned a rewriting rule in the format

$$P_i : \sigma_i \rightarrow \beta_{i,1} \beta_{i,2} \dots \beta_{i,14}, \quad (6.9)$$

where the successor of the rule consists of tokens $\beta_{i,1} \dots \beta_{i,14}$, which are represented by

$$\mathbf{y}_i = \left[x_{i,1} \ x_{i,2} \ \dots \ x_{i,14} \right]. \quad (6.10)$$

The successor is decoded from the vector $\mathbf{y}_i$ as:

- tokens $\beta_{i,1}$ and $\beta_{i,8}$:

$$\begin{aligned} \text{if } x_{i,j} \in [0, \frac{1}{2}], & \quad \beta_{i,j} = [\\ \text{else if } x_{i,j} \in [\frac{1}{2}, 1], & \quad \beta_{i,j} = \lambda \end{aligned}$$

- tokens $\beta_{i,2}$ and $\beta_{i,9}$:

$$\beta_{i,j} = \$ (g(x_{i,j}, \theta^{\min}, \theta^{\max}))$$

- tokens $\beta_{i,3}$ and $\beta_{i,10}$:

$$\beta_{i,j} = @ (g(x_{i,j}, c_s^{\min}, c_s^{\max}))$$

- tokens $\beta_{i,4}$ and $\beta_{i,11}$:

$$\beta_{i,j} = \& (g(x_{i,j}, c_w^{\min}, c_w^{\max}))$$

- tokens $\beta_{i,5}$, $\beta_{i,6}$, $\beta_{i,12}$ and $\beta_{i,13}$:

$$\text{if } x_{i,j} \in [0, \frac{1}{N_a+1}], \quad \beta_{i,j} = A$$

$$\text{if } x_{i,j} \in [\frac{1}{N_a+1}, \frac{2}{N_a+1}], \quad \beta_{i,j} = B$$

$$\text{if } x_{i,j} \in [\frac{2}{N_a+1}, \frac{3}{N_a+1}], \quad \beta_{i,j} = C$$

...

$$\text{if } x_{i,j} \in [\frac{N_a}{N_a+1}, 1], \quad \beta_{i,j} = \lambda$$

- tokens $\beta_{j,7}$ and $\beta_{j,14}$:

$$\text{if } \beta_{i,j-6} = [, \quad \beta_{i,j} =]$$

$$\text{if } \beta_{i,j-6} = \lambda, \quad \beta_{i,j} = \lambda,$$

where λ is an empty token. Further, g is a scaling function, defined as

$$g(x, c^{\min}, c^{\max}) = c^{\min} + x(c^{\max} - c^{\min}), \quad (6.11)$$

where $c^{\min}$ and $c^{\max}$ are the minimum and maximum bounds, respectively, of the design variable associated with a parametric symbol.

Before going into the encoding of the additional variables, let us introduce two new design variables. First, the non-dimensional extent variable is defined as

$$c_{\text{extent}} = l_{\text{branch}} / \sqrt{l^2 + (l/2)^2}, \quad (6.12)$$

where l_{branch} is the distance between the starting point of the turtle and the point in its path that is furthest away from the starting point (see Figure 6.8). The phenotypes are scaled in order to fit the parameter l_{branch} to satisfy Equation 6.12.

Second, the majority of the optimized results in the literature (e.g. the studies by Boichot and Fan (2016) and Marck et al. (2012)) consists of tree-like structures, where the width of the branches decreases when moving away from the heat sink. This supports the physical behavior involving branches becoming wider when approaching the heat sink, since they drive larger heat flux quantities collected through the domain. The parametric symbol $\&(c_w)$ enables changes in the prevailing width between steps, but not during a step. Therefore, we introduce a new variable $c_{t,j}$, specific to the letter σ_j in the alphabet, which changes the prevailing width during a step linearly from w_{i-1} to $w_i = w_{i-1}c_{t,j}$.

These variables induce the structural components of the phenotype to have a trapezoid shape, and thus we refer to them as trapezoid variables.

The last N_v elements of the vector $\mathbf{x}$ represent additional variables, which are

1. the vertical coordinate y_0 of the starting point of the turtle (see Figure 6.8),
2. the initial heading θ_0 of the turtle (see Figure 6.8),
3. the age n of the L-system,
4. the extent variable c_{extent} , and
5. trapezoid variables $c_{t,1} \dots c_{t,N_P}$.

The age n is an integer variable, encoded in the same way as the axiom letters (see the example in Equation 6.8), whereas the other additional variables are scalar variables, encoded via the scaling function g (Equation 6.11).

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

$$N_{\text{total}} = N_a + 14N_P + N_v, \quad (6.13)$$

where $N_v = 4 + N_P$.

In this study, we use an L-system design space, in which the axiom consists of four letters ($N_a = 4$), and the alphabet contains N_P letters, as well as the symbols described above. The number of encoded rewriting rules is equal to the number of letters in the alphabet. We define variables associated with the parametric symbols and additional variables to be bound between the minimum and maximum values listed in Table 6.1. Later, in Section 6.3.5, we study the influence of the number of letters in the alphabet, N_P , on the fitness of the optimized designs, as well as to the number of required objective function evaluations.

parameter	min	max
change in the heading θ (rad)	$-\pi/2$	$\pi/2$
relative change in step size c_s [-]	0.5	2.0
relative change in width c_w [-]	0.5	1.0
vertical coordinate y_0 (mm)	0	$l/10$
initial heading θ_0 (rad)	0	$\pi/2$
age n [-]	2	4
extent c_{extent} [-]	0.3	1.0
trapezoid variables $c_{t,i}$ [-]	0.4	1.0

Table 6.1: Minimum and maximum values of additional variables and variables associated with parametric symbols.

6.2.3 Mapping

The two previous sections explained our two parameterization methods of encoding two-dimensional path geometries, expressed using nodes, elements and non-dimensional element widths, into the vector $\mathbf{x}$. In this section, we describe the mapper (Figure 6.2), the main purpose of which is to scale the non-dimensional element widths, to satisfy the volume constraint, and project the resulting distribution of the high conductive material into a Cartesian design grid.

Before going into these, let us define how we transform elements into two-dimensional shapes. Depending on the parameterization method, each element is assigned one (in the ground structure approach) or two (in the L-systems-based method) non-dimensional widths. Thus, the resulting shapes are rectangles or isosceles trapezoids, respectively. As rectangles are also special cases of isosceles trapezoids, we refer to them as isosceles trapezoids too. Now, let us consider the latter and visualize the transformation of two elements into isosceles trapezoids and their projection into the Cartesian design grid in Figure 6.5(a). The dashed line shows the original elements between the nodes. The start and end widths of the first element are $c_{\text{sca}}w_i$ and $c_{\text{sca}}w_{i+1}$, respectively, where c_{sca} is the scaling factor.

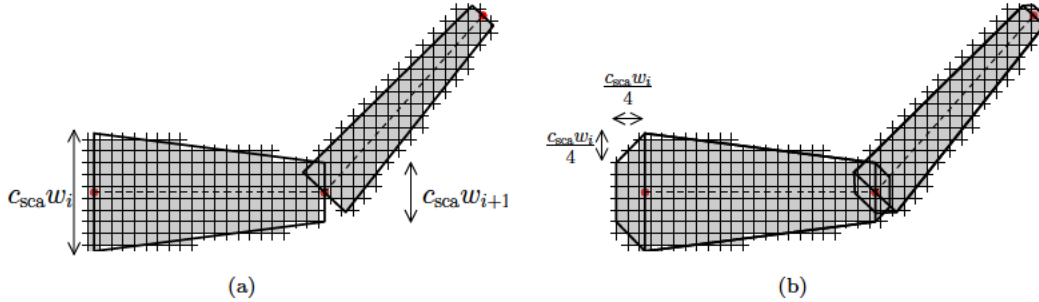


Figure 6.5: Projection of two example isosceles trapezoids, without (a) and with (b) additional isosceles trapezoids at their ends, into a Cartesian design grid.

As we can see from Figure 6.5(a), an alignment of two consecutive elements in an angle may induce the high conductive material distribution in the design grid to have a notch at the junction of the elements. Such notches drastically reduce the thermal performance of the structure. In order to prevent the formation of these notches, we define each isosceles trapezoid to have two additional isosceles trapezoids adjacent to their bases. Figure 6.5(b) shows the dimensions of these additional trapezoids, as well as their effect on the distribution of high conductive material in the Cartesian grid. Therefore, we construct each element as an octagon, which is the union of the main and two additional isosceles trapezoids.

In order to satisfy the volume constraint, we could scale the non-dimensional widths of these octagons by the scaling factor c_{sca} , so that their union⁶ E_u satisfies equation

$$|E_u(c_{\text{sca}})| = \phi|\Omega|. \quad (6.14)$$

However, using this scaling approach only would very rarely yield discretized designs with fully saturated volume constraint due to the following reasons. Both the ground structure approach and the L-systems-based method yield elements, of which some may touch the domain boundary⁷. As the octagon of such elements lies partly outside the domain Ω , the volume constraint is not saturated. In addition, the development of the L-systems branch is not restricted to the domain Ω (see the lower branch in Figure 6.6).

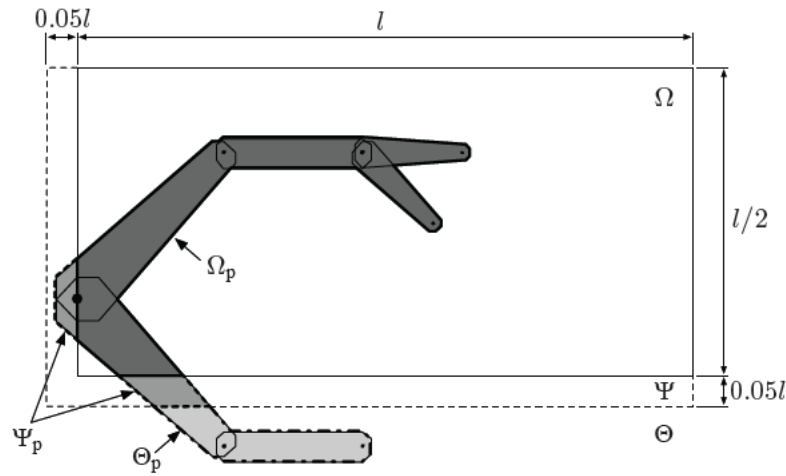


Figure 6.6: Visualization of the buffer domain Ψ located next to the south and west boundaries of the upper half of domain Ω . (This figure is prepared by Dr. Gilles Marck.)

To enable the saturation of the volume constraint, we define the buffer domain Ψ , adjacent to the south and west boundaries of the domain Ω (Figure 6.6), and domain Θ , such that $\Theta = \infty \setminus (\Omega \cup \Psi)$. Using the same analogy as with domains Ω and Ω_p , we define these two new domains to have subdomains Ψ_p and Θ_p , such that $\Psi_p = E_u \cap \Psi$ and $\Theta_p = E_u \cap \Theta$.

We determine the scaling factor c_{sca} via the following procedure:

1. find c'_{sca} , such that $|E_u(c'_{\text{sca}})| = \phi|\Omega|$
 2. find c_{sca} , such that $|\Omega_p(c_{\text{sca}})| = |\Omega_p(c'_{\text{sca}}) \cup \Psi_p(c'_{\text{sca}})|$
- (6.15)

Our approach to find the scaling factor c_{sca} naturally penalizes path geometries, yielded by the L-systems-based methods, which parts lie outside $\Omega \cup \Psi$. The volume constraint is saturated if, and only if, $\Theta_p(c'_{\text{sca}}) = \emptyset$.

⁶We compute the union of the octagons using the Python module Shapely (Gillies et al., 2007).

⁷By definition, the first element of our L-systems-based method always touches the boundary domain.

Since both areas $|E_u(c_{\text{sca}})|$ and $|\Omega_p(c_{\text{sca}})|$ increase monotonically as a function of the scaling factor c_{sca} , the steps in Equation 6.15 can be solved using a simple bisection method. We have chosen to use Brent's method, as we found it to be capable of finding the root for c_{sca} with acceptable precision in typically less than ten iterations.

Finally, the domain $\Omega_p(c_{\text{sca}})$ is projected onto the domain Ω , such that if the center of a design grid cell lies inside $\Omega_p(c_{\text{sca}})$ the cell belongs to domain Ω_p , and, if not, the cell belongs to domain Ω_0 . The projection method is described in detail in Appendix D.

6.2.4 Finite Volume Method (FVM)

The finite volume method (FVM) is a discretization method for the approximation of partial differential equations (PDEs). It is a well-established technology and applicable to various PDEs describing physical phenomena, such as fluid dynamics and heat transfer. We use the FVM to solve the temperature field of the domain Ω , governed by the PDEs in Equation 6.1.

Earlier, we discretized the high conductive material distribution in domain Ω into the Cartesian grid of $n_x \times n_y$ cells – here, we assign the corresponding thermal conductivity, k_p or k_0 , to the center point of these cells (Figure 6.7). In the FVM, two alternative schemes are typically used to discretize the finite volumes, which are referred to as the *centered* and *staggered grids*. In the former, the finite volumes are defined using the above mentioned Cartesian grid of $n_x \times n_y$ volumes, whereas, in the latter, the domain Ω is discretized into $(n_x + 1) \times (n_y + 1)$ volumes and the boundary volumes are defined to have half the size of the other cells (Figure 6.7).

Later in this chapter, we will benchmark our results against the SIMP method, which typically is used in conjunction with the staggered grid to avoid the checkerboarding problem (see Sections 2.5 and 2.6.1). Therefore, to enable a fair comparison, we choose to use the staggered grid also with the ground structure approach and L-systems-based method.

The FVM requires some average of the conductivity at the interface of two finite volumes (see $k_{i+\frac{1}{2}}$ in Figure 6.7), to evaluate the heat flux between the finite volumes, as well as, in the case of the staggered grid, a representative heat generation rate of a finite volume⁸. In the literature, two averages are used to describe these quantities, which are

$$\begin{aligned} \text{arithmetic (Voigt) average : } \bar{k} &= \frac{1}{n} \sum_{i=1}^n k_i \\ \text{harmonic (Reuss) average : } \bar{k} &= n \left(\sum_{i=1}^n \frac{1}{k_i} \right)^{-1} \end{aligned} \tag{6.16}$$

⁸In Section 6.1, we eliminated the heat generation of the high conductive material domain Ω_p

We use the arithmetic average, with $n = 2$, to estimate the conductivity at the finite volume interface, using the two conductivities assigned to the end points of the finite volume interface. We also use the arithmetic average with $n = 4$ to represent the heat generation rate in a finite volume, using the four heat generation rates assigned to the corners of the finite volume.

Figure 6.8 summarizes the process of scaling, projecting and evaluating the temperature distribution of a phenotype. In terms of the computational cost, the projecting step requires approximately the same run time as solving the temperature field. We studied several design grid densities and found that the grid of 200×100 finite volumes yielded a good trade-off between the solution accuracy and the computational cost. The total run time of evaluating a genotype, encoded using the ground structure approach or L-systems-based method, was typically around 0.2 to 4 seconds on a single Central Processing Unit (CPU), depending on the complexity of the corresponding phenotype. We have observed that the longest run times typically occur during the first generation of the GA process. In the remaining of this chapter, we evaluate generations of individuals in parallel using 16 CPUs.

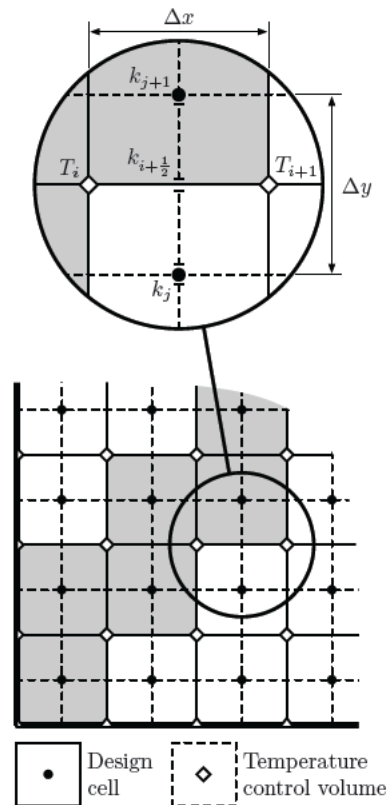


Figure 6.7: Staggered finite volume grid. (This figure is prepared by Dr. Gilles Marck.)

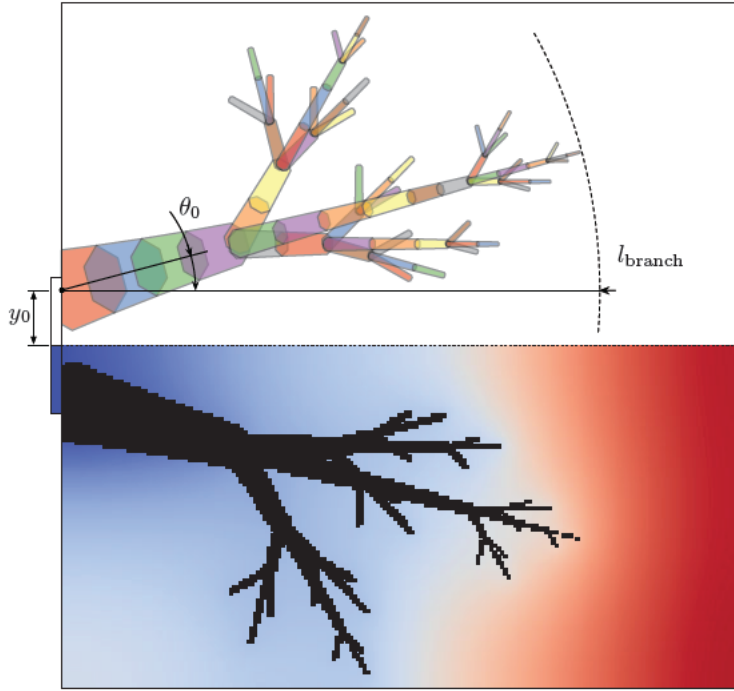


Figure 6.8: An illustration of scaling (upper part), projection (lower part) and evaluating the temperature distribution (lower part) of the slightly modified version of the parametric L-system, plotted earlier in Figure 2.8(d). We have modified it by introducing the trapezoidal variables (see Section 6.2.2)

6.3 Results

In this section, first, we study which of the proposed parameterization methods, i.e. the ground structure approach or the L-systems-based method, yields better results on a single-objective topology optimization problem, characterized by optimization parameters listed in Table 6.2.

We found that, out of the two methods, L-systems-based method yielded clearly better results. Therefore, second, we benchmark it against the most relevant studies in the literature on six single-objective optimization problems, the optimization parameters for which will be presented later in Section 6.3.4. Finally, we demonstrate the suitability of the L-systems-based method to tackle multi-objective problems with relevant temperature and manufacturing related objectives.

parameter	value	unit
heat generation rate q_p	0	kW/m ²
heat generation rate q_0	10	kW/m ²
length l	0.1	m
dimension d	$0.2l$	m
conductivity k_p	50	W/(mK)
conductivity k_0	1	W/(mK)
volume fraction ϕ	0.1	-
objective function $f(\chi_{\Omega_p})$	$\bar{T}$	K

Table 6.2: Parameters defining the optimization problem.

6.3.1 Control parameters in the ground structure approach

Let us start by examining which GA control parameters to use with the ground structure approach. In Chapter 3, we studied the effects of the population size and crossover type on the performance of the GA. We selected the best performing parameters for the application of the sUAV wing internal structure. Further, in Chapter 4, we studied the effects of a more extensive set of 432 control parameter combinations on the performance of the GA on five test cases. However, this experiment was conducted on the map L-systems-based parameterization, which is fundamentally different to the ground structure approach.

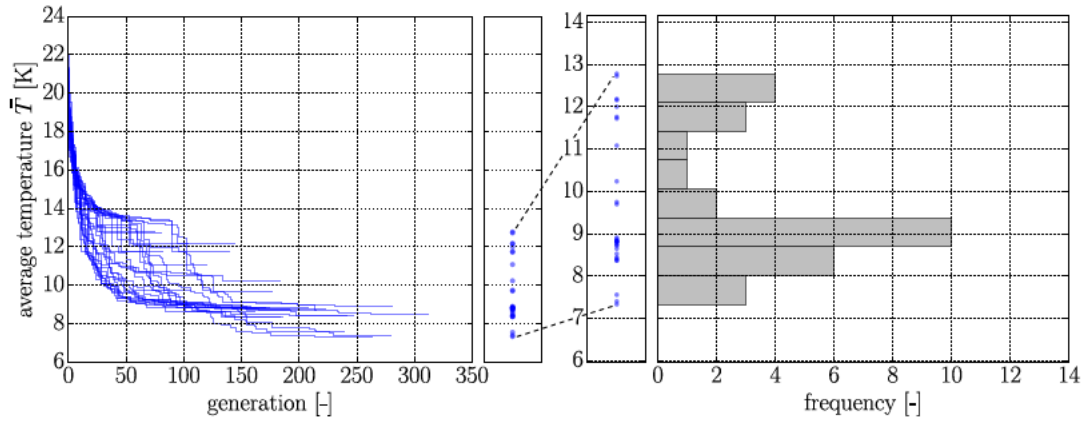
The question here is whether we can exploit the results of the statistical experiment (on a different parameterization method) to improve the performance of the GA on the ground structure approach. To investigate this, we run experiments with both the control parameters⁹ from Chapter 3 and the control parameter combination #4 from Table 4.4. The latter was chosen from Table 4.4 because it has the same population size $N_{\text{pop}} = 150$ as the former.

One of the conclusions of Chapter 3 was that, on the specific application and parameterization method, GAs with the distributed crossover have better performance than those with the two-point crossover. The crossover type of the parameter combination #4 is the two-point crossover. Therefore, we also run experiments with a modified version of the parameter combination #4, where we set the crossover type to be the distributed crossover.

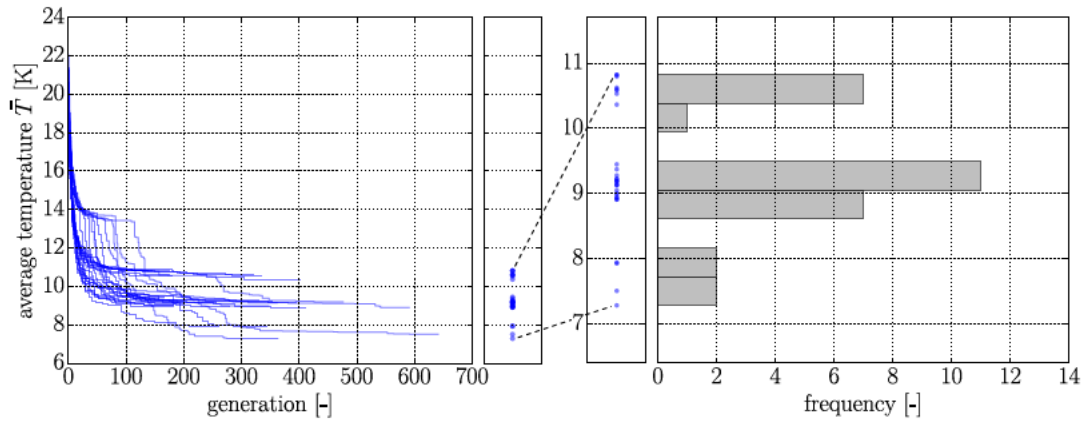
The optimization problem is characterized by the parameters listed in Table 6.2. We use a quadrilateral ground structure with diagonals, with a density of 12×6 , and choose the number of non-dimensional widths to be $N_w = 2$. We repeat the optimization processes 30 times with each of the three parameter combinations, and we terminate the optimization processes when no improvements are found during 50 consecutive generations.

⁹These control parameters are: $N_{\text{pop}} = 150$, $N_{\text{pool}} = 4$, $c_x = 0.9$, $c_m = 0.02$, $X_{\text{type}} = \text{'distributed'}$ and $E_{\text{bool}} = \text{True}$.

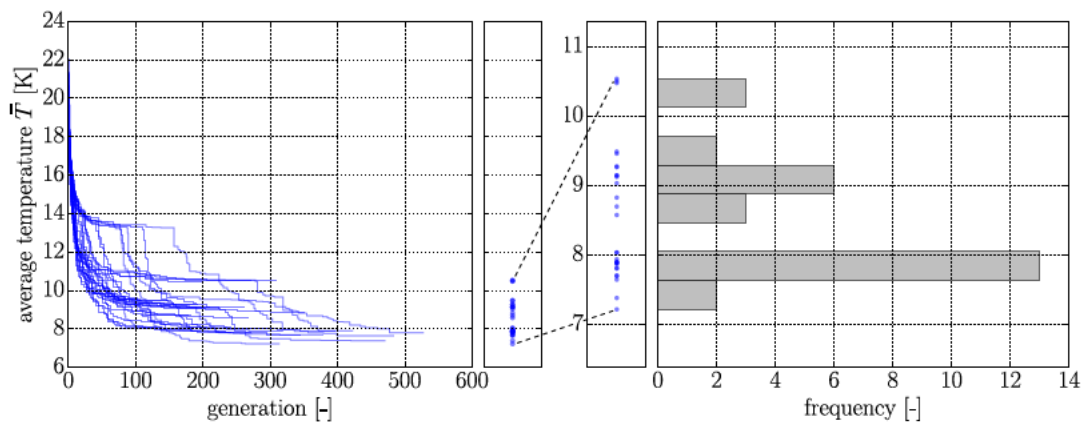
Figure 6.9 shows the convergence histories and distributions of the optimized average temperature $\bar{T}$ and, further, Figure 6.10 summarizes the results of the three algorithms with different control parameters.



(a) control parameter combination from Chapter 3



(b) control parameter combination #4



(c) control parameter combination #4 (with the distributed crossover)

Figure 6.9: Convergence histories and distributions of optimized average temperature $\bar{T}$ of the ground structure approach. Optimization runs with each of the three studies control parameter combinations are repeated 30 times.

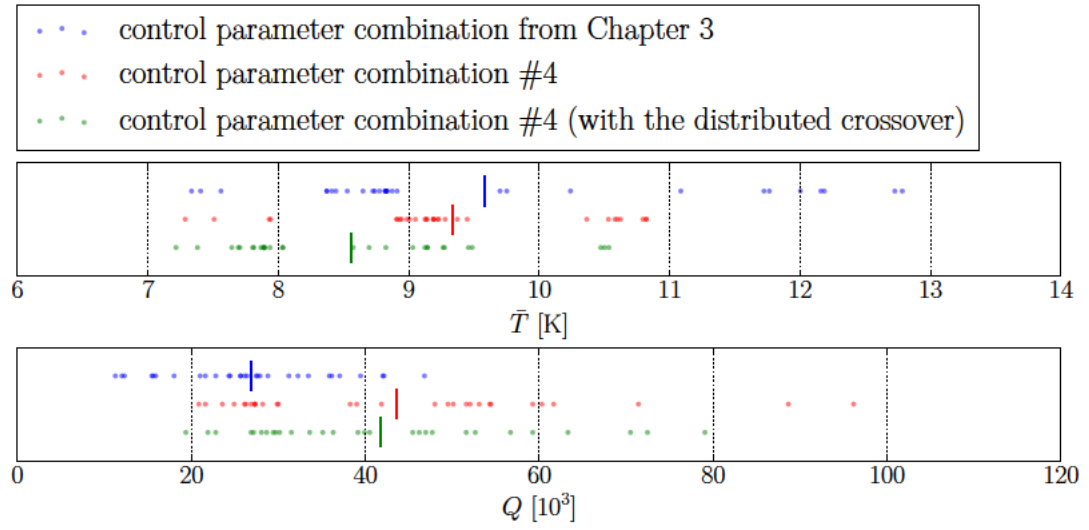


Figure 6.10: Distributions of the optimized average temperature $\bar{T}$ and the number of objective function evaluations Q with the three tested control parameters combinations. The data sets contain 30 repeats, and their averages are indicated by vertical lines.

We test the statistical significance of these results by a family of six statistical tests, which includes the pairwise comparisons of the average temperature $\bar{T}$ and the number of objective function evaluations Q . This family of statistical tests is similar to the one we conducted in Section 3.7.4.4. The null hypothesis H_0 is that the means of the tested quantity, $\bar{T}$ or Q , are similar ($\mu_1 = \mu_2$) and the alternative hypothesis H_1 is that they are dissimilar ($\mu_1 \neq \mu_2$). We again use the level of significance $\alpha = 0.05$, and threat the inflated probability of Type I error by using the step-up false discovery rate (FDR) (Equation 3.9).

Based on Figures 6.9 and 6.10, both distributions of the optimized average temperature $\bar{T}$ and the number of objective function evaluations Q are multimodal. Thus, we determine the p values using the nonparametric Mann-Whitney U-test.

Table 6.3 shows the results of the statistical tests, which are ranked in the decreasing order of significance. Starting from the highest p value, the first test satisfying Equation 3.9 is ranked 4th. Thus, we reject the null hypothesis H_0 in tests $j = 1, 2, \dots, 4$, and accept it in tests $j = 5, 6$.

As a result, we obtain statistical significance that the algorithm with the control parameter combination from Chapter 3 requires fewer objective function evaluations than those with the two other control parameter combinations. However, this may be a consequence of the algorithm converging repeatedly prematurely to local optima (see Figure 6.9(a)). More importantly, we also obtain statistical significance that the algorithm with the modified version of control parameter combination #4 yields lower average temperatures than the algorithms with the two other control parameter combinations.

Because of this, we will use it with the ground structure approach in the remaining of this chapter.

rank j	quantity	control parameters 1	control parameters 2	U	p	$j\alpha/n$
1	Q	Chapter 3	#4 (distributed)	184.5	7.51E-05	8.33E-03
2	Q	Chapter 3	#4	211.0	2.11E-04	1.67E-02
3	$\bar{T}$	#4 (distributed)	#4	245.0	2.03E-03	2.50E-02
4	$\bar{T}$	#4 (distributed)	Chapter 3	289.0	1.37E-02	3.33E-02
5	$\bar{T}$	#4	Chapter 3	375.0	1.35E-01	4.17E-02
6	Q	#4	#4 (distributed)	433.5	4.94E-01	5.00E-02

Table 6.3: The family of statistical tests ($j = 1 \dots 6$), ranked in decreasing order of significance. In each test, control parameters 1 has on average of the obtained results a better performance than control parameters 2. Based on the results, we reject the null hypothesis H_0 in tests $j = 1 \dots 4$.

6.3.2 Ground structure type and density

In this section, we study the effects of the type and density of the ground structure on the optimized average temperature $\bar{T}$ and the number of objective function evaluations Q . We retain the other optimization parameters the same as in the previous section. The types of the ground structure are the quadrilateral with diagonals (Figure 6.4(a)) and hexagonal (Figure 6.4(b)), and we use densities 4×2 , 8×4 and 12×8 . These ground structures are fitted to the domain Ω so that at least one of the ground structure members touch the center of the heat sink.

We again repeat the optimization runs with each of the six ground structures 30 times. Figure 6.11 shows the distributions of optimized average temperature $\bar{T}$ and the required objective function evaluations Q , and Figure 6.12 visualizes the best obtained designs.

With both ground structure types, the average number of required objective function evaluations clearly increases when we increase the density of the ground structure. However, the best designs, on average, are rather surprisingly obtained with the coarsest quadrilateral ground structure (density 4×2). The designs obtained with this ground structure have, on average, 34.7% lower average temperature $\bar{T}$ than the designs obtained with the same type of ground structure but with density 8×4 . Considering that the design space of the latter contains all designs of the former, the result indicates that the finer quadrilateral ground structures (densities 8×4 and 12×6) have design space complexities beyond the capabilities of the GA.

However, the quadrilateral ground structure with density 12×6 yields, on average better designs than with density 8×4 . We believe that the reason is the fact that in the former, unlike the latter, contains ground structure members, which touch the heat sink at more than one locations (see Figures 6.12(b) and 6.12(c)).

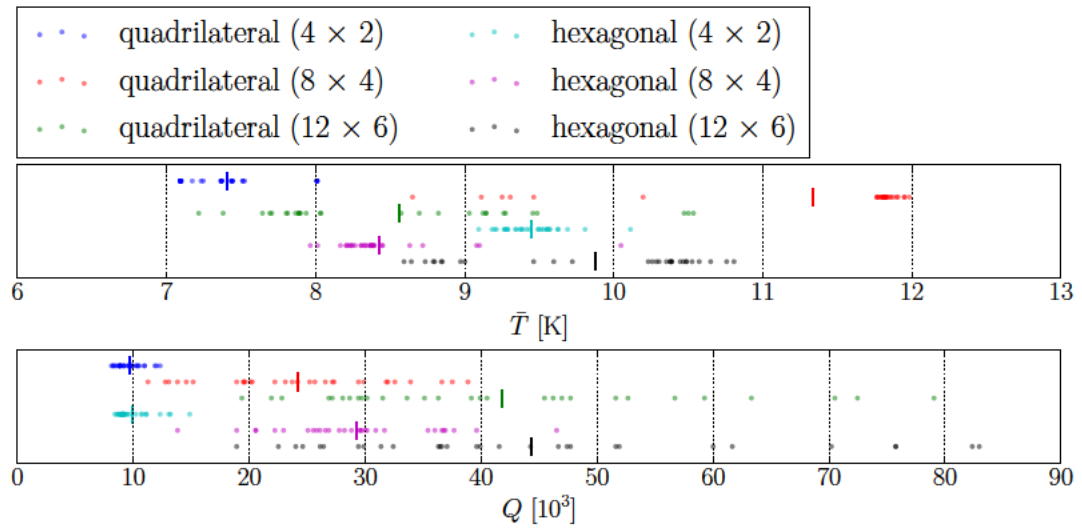


Figure 6.11: Distributions of optimized average temperature $\bar{T}$ and number of objective function evaluations Q of optimization runs initiated from quadrilateral (with diagonals) and hexagonal ground structures with various densities. The data sets contain 30 repeats, and their averages are indicated by vertical lines.

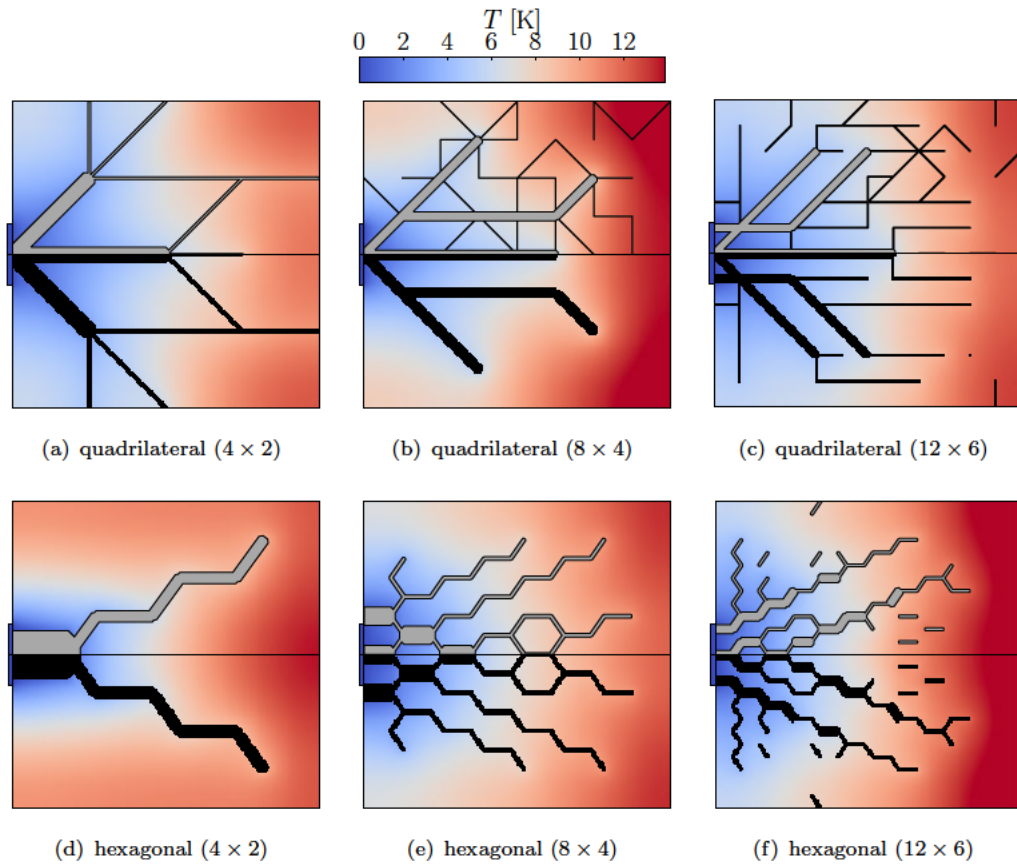


Figure 6.12: Best designs obtained using different densities of quadrilateral (with diagonals) and hexagonal ground structures.

The coarsest hexagonal ground structure (density 4×2) has roughly the same design space size as the coarsest quadrilateral ground structure (the numbers of design variables are 21 and 23, respective). However, the designs obtained with the former have, on average, 27.6% higher average temperature $\bar{T}$ than with the latter. The reason could be that the coarsest hexagonal ground structure only has a single ground structure member touching the center point of the heat sink, while in the corresponding quadrilateral ground structure the corresponding number is three. Considering only the hexagonal grounds structures, the best designs, on average, are obtained with the density 8×4 .

We conclude that the optimized average temperature $\bar{T}$ depends significantly on the used ground structure and, yet, the type and density of the most suitable ground structure are difficult to determine prior the optimization process.

6.3.3 Ground structure approach versus L-systems-based method

The L-systems-based method, which we described in Section 6.2.2, does not require such a priori information. In this section, we use it to solve the above defined optimization problem.

Let us next choose the control parameters for the GA that evolves the L-system designs. In Chapter 4, we performed a statistical experiment of various control parameters on the performance of the map L-systems-based method. However, because the interpretation formalism of L-systems we use here (the turtle interpretation) is not the same as in the statistical experiment (map L-systems), the obtained results may not be applicable when choosing the control parameters for the current application. Nevertheless, as both parameterizations are still based on L-systems and we lack of a better guess of the most suitable control parameters, we rely on the results obtained from the statistical experiment. Thus, we pick the control parameter combination #4 (Table 4.4), which has the same population size $N_{\text{pop}} = 150$ as in the ground structure approach.

However, because the current parameterization contains many scalar variables, we here change the mutation operator to a Gaussian mutator¹⁰, with mean $\mu = 0$ and standard deviation $\sigma = 0.3$. This mutation operator may set a real variable of the vector $\mathbf{x}$ (Equation 6.7) outside its bounds $[0, 1]$, in which case we repair it by adding/subtracting the appropriate integer number, e.g. -0.1 becomes 0.9 .

We define the L-system alphabet Σ to consist of letters $\{A, B, C, D\}$ (as well as the symbols described earlier) and, thus, the number of encoded rewriting rules is $N_P = 4$. We terminate the optimization runs when no improvement is found during 50 consecutive generations.

¹⁰Swap mutator was used in the statistical analysis.

Figure 6.13 shows the convergence histories and distributions of the optimized average temperature $\bar{T}$ of 30 repeated optimization runs. In comparison to the corresponding plots with the ground structure approach (Figure 6.9), these histories and optimized average temperatures are more consistent with each other. 26 out of 30 optimization runs yield designs which average temperature $\bar{T}$ is within 3.8% of that of the best design.

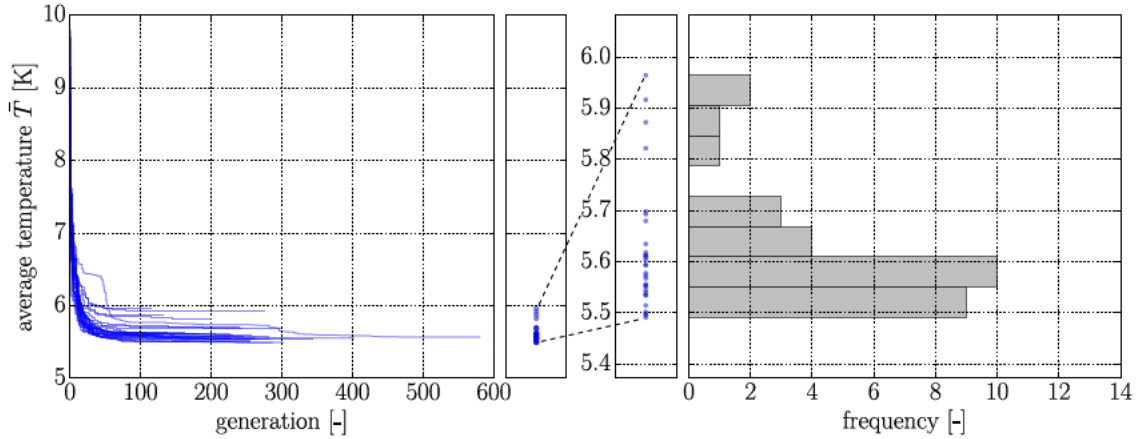


Figure 6.13: Convergence histories and distributions of optimized average temperature $\bar{T}$ of 30 optimization runs with the L-systems-based method.

Figure 6.14 presents a comparison of the obtained results to those obtained using the quadrilateral ground structure of the density 4×2 , which in our experiment yielded the best designs. The designs obtained by the L-systems-based method have, on average, 24.1% lower average temperature $\bar{T}$ than those obtained by the ground structure approach. However, the ground structure approach requires fewer objective function evaluations than the L-systems-based method.

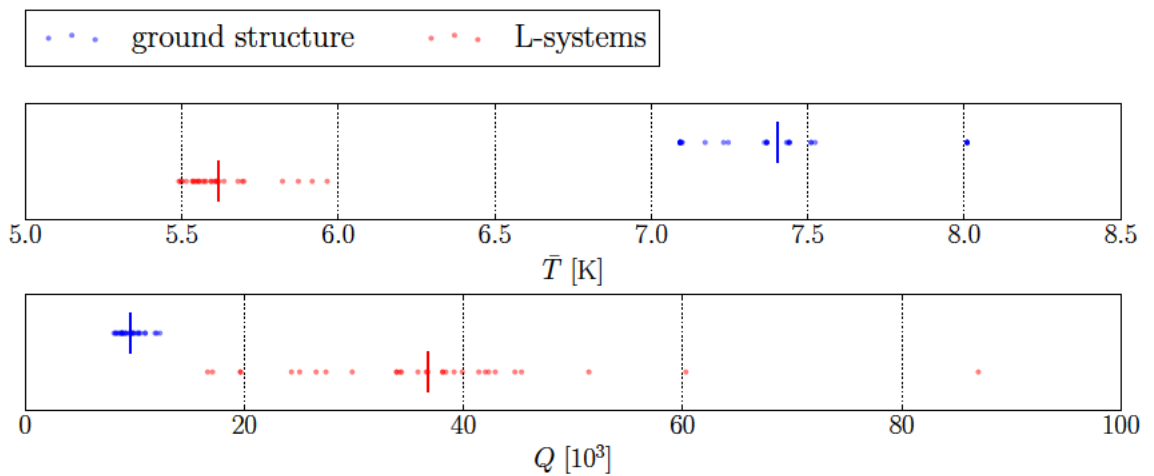


Figure 6.14: Comparison of the optimized average temperature $\bar{T}$ and the number of objective function evaluations Q between the ground structure approach and the L-systems-based method.

Finally, let us examine the best design obtained by the L-systems-based method (Figure 6.15). The high conductive material in this design is distributed in radially aligned branches, which initiate from the heat sink. The widths of these branches decrease gradually towards their tips. From a physical point of view, the radially aligned branches effectively direct the heat flux towards the heat sink in the entire high conductive material domain Ω_p . In addition, the total heat flux in these branches is the greatest near their roots, and decreases towards their tips. Thus, the decreasing width of the spikes also makes sense from the physical point of view.

The designs obtained using the hexagonal ground structure resemble to some extent this formation of radial branches – with the difference that their branches have zigzagging shapes. Arguably, we could obtain better designs with the ground structure approach if we would align the ground structure members in radial patterns. However, such alignment may no longer be favorable if we vary the optimization problem parameters, listed in Table 6.2, in which case we would be required to again iteratively search the suitable ground structure.

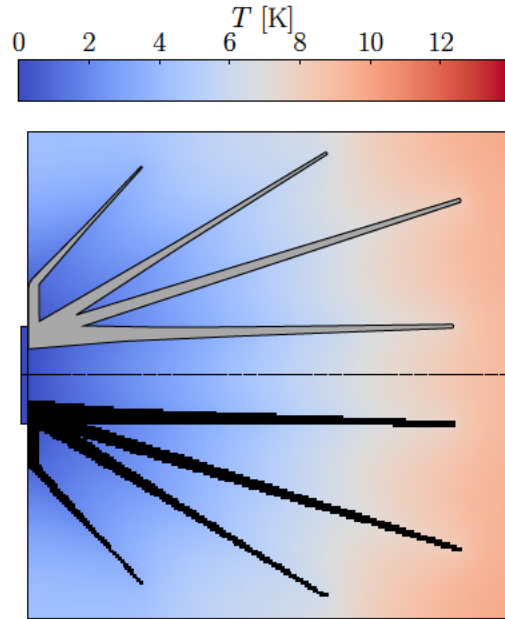


Figure 6.15: The best design obtained using the L-systems-based method. The range of the contour map is the same as in Figure 6.12.

In the remaining part of this chapter, we will focus on the L-systems-based method, because it yielded optimized designs with clearly lower average temperatures $\bar{T}$ than the ground structure approach on the studied optimization problem, and does not require such a priori definition of the candidate structural members as the ground structure approach. In the following, we benchmark the method against relevant studies in the literature on several single-objective optimization problems, as well as demonstrate the

method in multi-objective optimization with relevant temperature and manufacturing related constraints.

6.3.4 Boichot's problem parameters

Let us start by selecting relevant reference studies in the literature. As we mentioned in the Section 2.8.2, Boichot and Fan (2016) showed that their GA-based algorithm yields lower non-dimensional thermal resistances than the studies using cellular automata, constructal theory and ESO (see references in Section 2.8.2). Let us refer to this approach as the direct encoding method. The non-dimensional thermal resistance, specific to a reference temperature T_{ref} , is defined as

$$R_{\{T_{\text{ref}}\}} = \frac{T_{\text{ref}} - T_{\text{sink}}}{q_0 A / k_0}, \quad (6.17)$$

where T_{sink} is the temperature of the heat sink, q_0 is the heat generation rate within the domain Ω_0 , and A is the area of the domain¹¹ (Bejan, 1997). The reference temperature T_{ref} is either $\bar{T}$ or T_{max} , depending on the objective function studied. The purpose of the non-dimensional thermal resistance is to enable the comparison of optimized designs with different problem parameters q_0 , A , k_p/k_0 or ϕ .

Boichot and Fan (2016) also indicate that their results are similar to those obtained by Marck et al. (2012) using the SIMP method. Therefore, we choose to benchmark our results against these two studies.

Looking more closely into the comparison by Boichot and Fan (2016), the comparability of non-dimensional thermal resistances between these two studies is, in fact, limited, due to the following reasons. First, Marck et al. (2012) define the heat generation to occur in both domains Ω_0 and Ω_p , whereas Boichot and Fan (2016) set it only to the domain Ω_0 . Second, the heat sinks have different sizes; Marck et al. (2012) define it to be 1% of the left boundary, whereas Boichot and Fan (2016) define it to be 20% (of the same boundary). Third, discretizations of the design domains are different. Marck et al. (2012) used a staggered grid of 200×100 elements, whereas Boichot and Fan (2016) used a centered grid of 100×50 elements.

To ensure a fair comparison between the three methods, we test our L-systems-based method, presented in this work, on the same optimization problems that were studied by Boichot and Fan (2016) and generate the corresponding results using the SIMP method¹² implemented by Marck et al. (2012). The optimization problems are defined based on the objective function, the conductivity ratio k_p/k_0 and the volume fraction ϕ

¹¹In our case, the design domain has a square shape, and therefore $A = l^2$.

¹²In the SIMP method, we use a sensitivity filter with a radius of $1.25\Delta x$ (or $1.25\Delta y$) to avoid checkerboarding.

(see Table 6.4), while the other problem parameters, q_p , q_0 , l , d and k_0 , are fixed to the values defined in Table 6.2.

Depending on the method under consideration, the design grid does not have the same size:

- as mentioned earlier, [Boichot and Fan \(2016\)](#) generated the direct encoding results using the centered grid of 100×50 elements,
- in the analysis using the L-systems-based approach, we use a staggered grid of 200×100 elements, which provides a suitable trade-off between the design accuracy and fast mapping,
- SIMP method uses a grid of 400×200 elements. Indeed, this approach required a filtering step in order to avoid the so-called *checkerboard* problem, which artificially aggregates high-conductivity elements together. Consequently, the thinnest branches that the SIMP method is able to produce have the same width as the one coming from the L-system approach, ensuring a meaningful comparison between both designs.

Finally, we evaluate all the optimized designs (including the ones obtained by [Boichot and Fan \(2016\)](#)) using the same staggered grid of 800×400 elements and the same FVM solver (see Section 6.2.4). This involves an additional projection method, which we use to map the designs from the coarse grids (that hold 100×50 , 200×100 or 400×200 elements) to the fine grid of 800×400 elements.

$f(\chi_{\Omega_p})$	#	k_p/k_0 [-]	ϕ [-]	#	$f(\chi_{\Omega_p})$
$\bar{T}$	1	2	0.3	7	$T_{\max}$
	2	10	0.1	8	
	3	10	0.3	9	
	4	10	0.5	10	
	5	50	0.3	11	
	6	250	0.3	12	

Table 6.4: Optimization problem objectives and parameters k_p/k_0 and ϕ : in optimization problems #1-6 the objective is to minimize the average temperature $\bar{T}$, whereas in optimization problems #7-12 the objective is to minimize the maximum temperature $T_{\max}$.

6.3.5 Variation of the L-systems-based method parameters

Before generating the results with the L-systems-based method, let us conduct trade-off studies on two main parameters of the method. We conduct the trade-off studies on the optimization problems #2 and #6 (Table 6.4), which have both different conductivity

ratios k_p/k_0 and volume fractions ϕ . We repeat the optimization runs 30 times with each of the studied parameters.

6.3.5.1 Control parameters

The first parameter is the choice of control parameter combination from Table 4.4, which represents the Pareto front in the space of average optimized objective function value and the average number of objective function evaluations. As we approximated the Pareto front using five low-cost test cases, an interesting question here is whether we would obtain similar results on the current ‘real-life’ topology optimization problem. If positive, what is the trade-off between the optimized average temperature $\bar{T}$ and the number of objective function evaluations Q on the current optimization problem.

We repeated optimization runs on both optimization problems with control parameter combinations #1, #4, #7 and #12, which represent points at the Pareto front with different population sizes, i.e. 200, 150, 100, and 50, respectively. Figure 6.16 presents the distributions of the optimized average temperature $\bar{T}$ and the number of objective function evaluations, Q , of these optimization runs.

In the optimization problem #2, the average number of required objective function evaluations, $\bar{Q}$, has the same trend as in the Pareto front obtained from the statistical experiment: the order of the control parameter combinations is #12, #7, #4, #1, starting from the one that defines the algorithm with the smallest computational cost. Based on the results, we can also see that the spread of Q increases significantly with the computational cost.

However, the average of the optimized average temperature $\bar{T}$ of the control parameter combination #4 seems to be lower than that of the combination #1, which is against the expected trend. To see whether the trend continues, we amended the tested control parameter combinations with a modified combination #1, in which the population size is increased¹³ to $N_{\text{pop}} = 300$. As a result, the unexpected trend continued (see Figure 6.16(a)). On the other hand, the averages of the optimized average temperature $\bar{T}$ and the number of objective function evaluations Q follow the expected trends in the optimization problem #6. Although we do not know the reason for the unexpected trend and acknowledge that our sample size is still fairly small, we select the control parameter combination #4 for the remainder of this chapter.

¹³Earlier, we found that, on the test cases, the increase in the population size N_{pop} increases both the average fitness of the optimized designs and the average number of objective function evaluations (see Figure 4.4(a) and 4.5(a)).

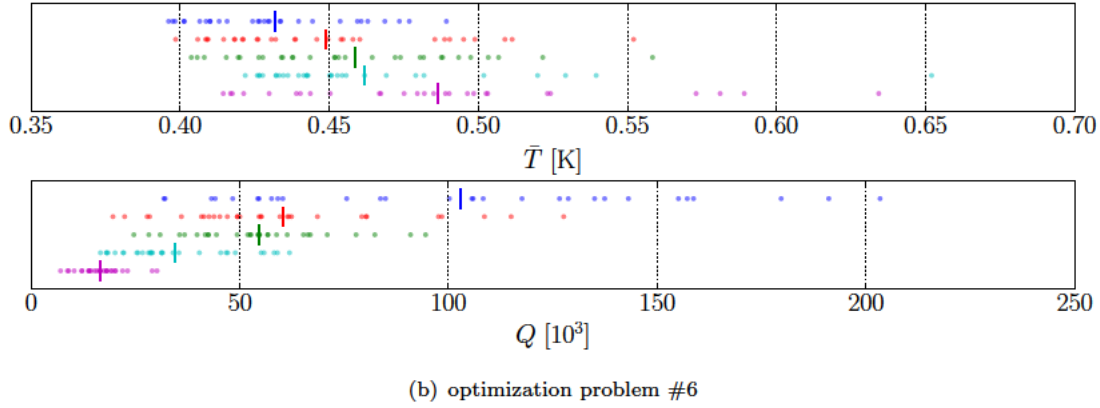
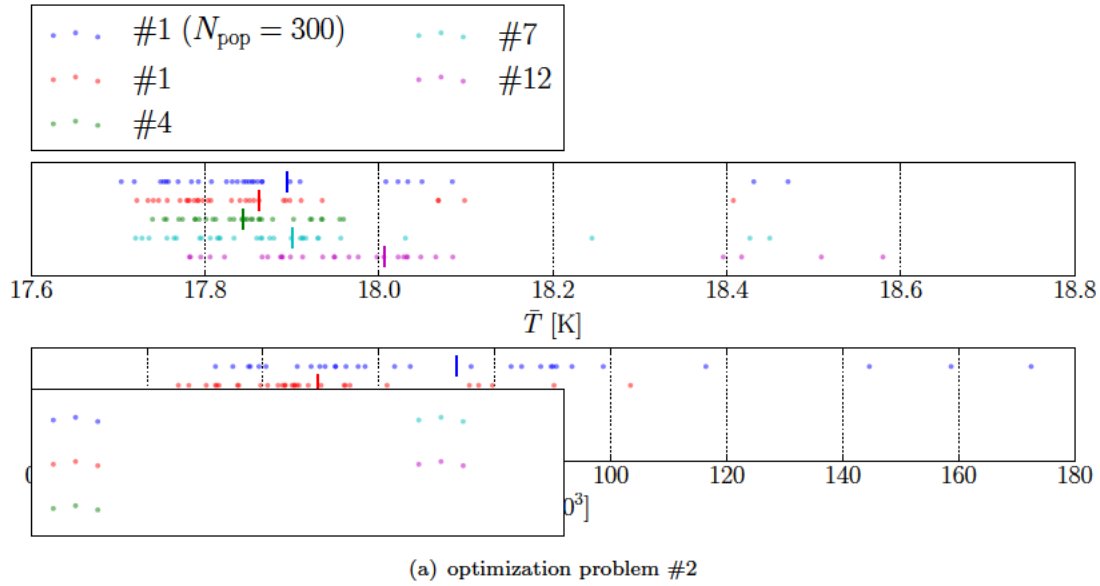


Figure 6.16: Trade-off study of the control parameter combinations on the optimized average temperature $\bar{T}$ and the number of objective function evaluations Q .

6.3.5.2 Number of encoded rewriting rules

The second parameter, we have chosen to study, is the number of encoded rewriting rules, N_p , which is one of the main parameters representing the size of the design space in the L-systems-based method. Another interesting parameter representing the size of the design space would be the length of the axiom, but we limit the trade-off study to the former only.

Presumably, the more rewriting rules are encoded, the better is the best design in the design space and the harder it is to find (i.e. the more objective function evaluations are required). Let us refer to this as the expected trend.

We repeated the optimization processes on the two optimization problems using the numbers of rewriting rules $N_P = \{2, 4, 6\}$. Figure 6.17 shows the results of these optimization runs. In both optimization problems, the algorithm with $N_P = 2$ yielded, on average, the worst optimized designs and required the smallest number of objective function evaluations. This behavior follows the expected trend.

The algorithm with $N_P = 6$ yielded, on average, slightly better designs than the algorithm with $N_P = 4$ in optimization problem #6 (the expected trend); however, the opposite result was obtained in the optimization problem #2. Considering the large deviation in their numbers of objective function evaluations Q , we can only conclude that computation costs of these algorithms seem similar.

In the remaining of this chapter, we encode four rewriting rules ($N_P = 4$). Thus, the alphabet Σ contains letters $\{A, B, C, D\}$, as well as the symbols described in Section 6.2.2.

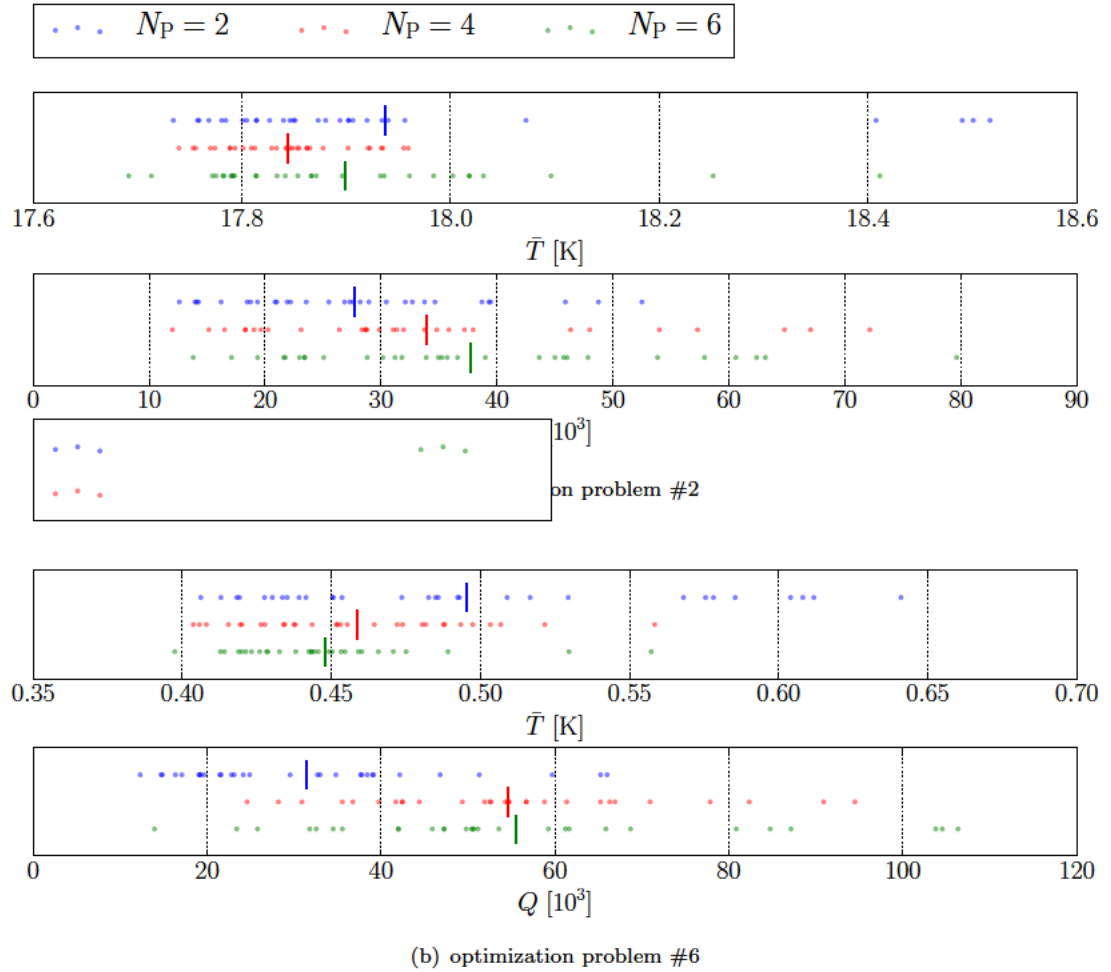


Figure 6.17: Trade-off study of the number of encoded rewriting rules, N_P , on the optimized average temperature $\bar{T}$ and the number of objective function evaluations Q .

6.3.6 Benchmarking on single-objective optimization problems

Figure 6.18 shows the design comparison of optimization problems #1-6, where the objective is to minimize the average temperature $\bar{T}$. In these optimization problems, the SIMP method (Marck et al., 2012) yields optimized designs with 2.7% to 20.6% lower average temperatures than the direct encoding method (Boichot and Fan, 2016). Therefore, we normalize all results in the figure with respect to those obtained by the SIMP method. In fact, the results of the direct encoding method become increasingly worse in comparison to the SIMP method when $\phi k_p/k_0$ increases. The corresponding numerical values are listed in Table 6.5.

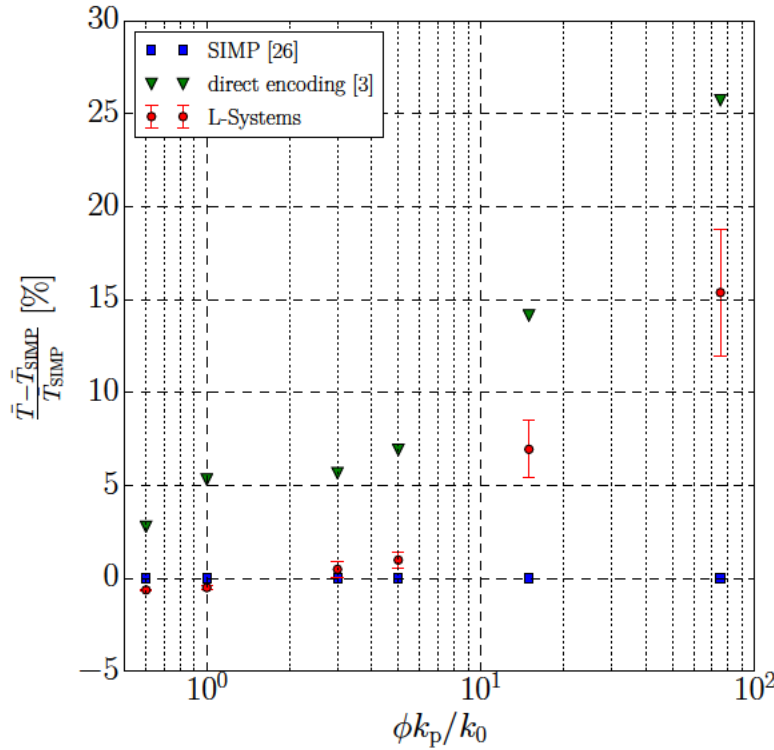


Figure 6.18: Benchmarking of the L-systems-based method against the direct encoding (Boichot and Fan, 2016) and the SIMP method (Marck et al., 2012) for problems #1-6. $\bar{T}_{\text{SIMP}}$ is the average temperature optimized by the SIMP method.

As the L-systems-based method is stochastic¹⁴, we have repeated the optimization runs 30 times for each problem. This time the optimized average temperatures are normally, or nearly-normally distributed, instead of having such multi-modal distribution that we saw with the ground structure approach. All distributions are shown in Appendix E (Figure E.1). Therefore, we report the mean of the optimized objective function values and its 95% confidence interval, calculated by multiplying the standard error by 1.96 (see Figure 6.18 and Table 6.5).

¹⁴The GA-based algorithm by Boichot and Fan (2016) is also stochastic, but they only provide between one and three solutions for each case.

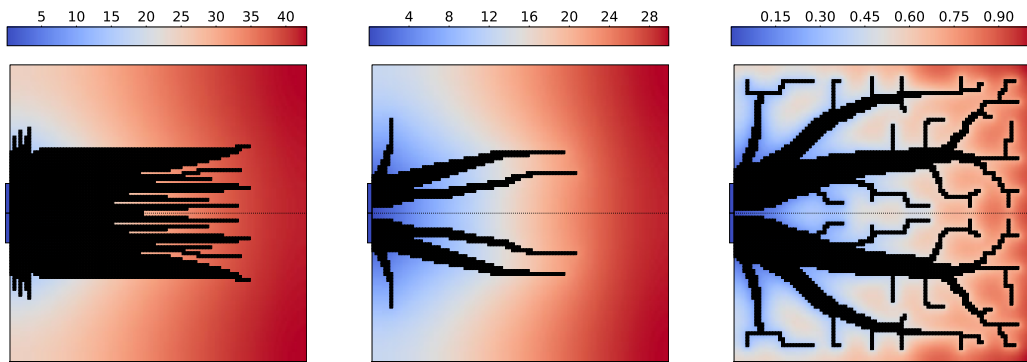
The L-systems-based method yields on average better designs than the direct encoding method (Boichot and Fan, 2016) in problems #1-6. These conclusions are statistically significant.

The effectiveness of L-systems-based method against the SIMP method seems to be dependent on the dimensionless coefficient $\phi k_p/k_0$ (Figure 6.18). When $\phi k_p/k_0 \leq 1$ (optimization problems #1-2), the L-systems-based method yields lower objective function values than the SIMP method, whereas, when $\phi k_p/k_0 \geq 3$ (optimization problems #3-6), the optimized objective function values are higher. Looking at the optimized designs in Figure 6.19, the complexity of the designs seems to be related to the dimensionless coefficient $\phi k_p/k_0$. We name two potential reasons why the L-systems-based method cannot find as good designs as the SIMP method in optimization problems where $\phi k_p/k_0 \geq 3$: 1) the parameterization is not flexible enough to define designs with required geometrical complexity (see Figure 6.19(f) as a reference) and/or 2) the method fails to fine-tune the details of these designs as it does not use the gradient information of the objective function.

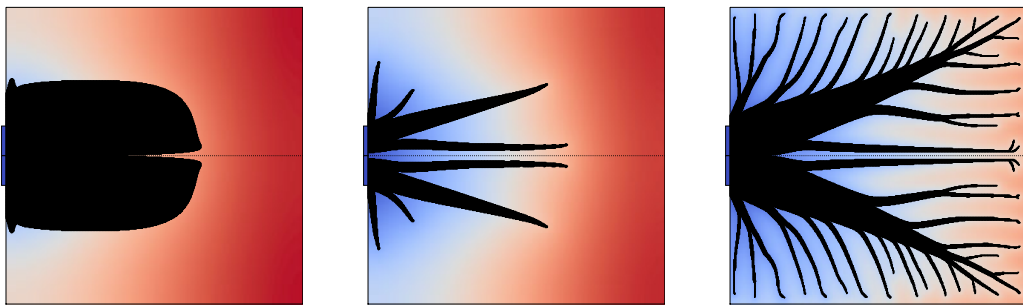
Figure 6.19 presents a comparison of optimized designs for three representative optimization problems (#1, #2 and #6)¹⁵. The optimized designs for the other optimization problems (#3, #4 and #5) are presented in Figure F.1 in Appendix F.

In optimization problem #1 (Subfigures a, d, g), the L-systems-based method yields a design where the North and South boundaries of the high conductive material are clearly coarser than in the reference designs.

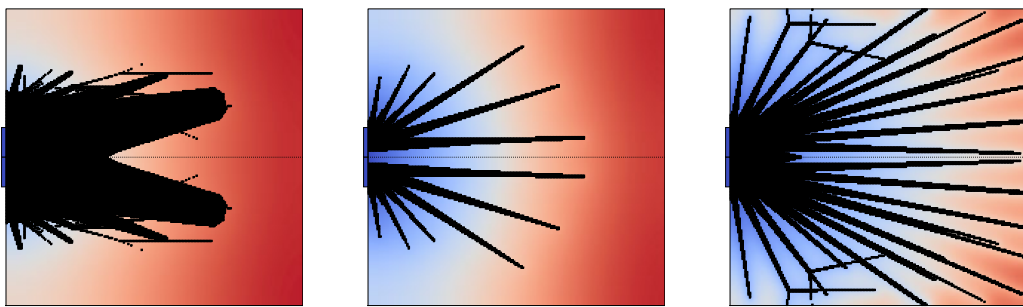
¹⁵The presented designs corresponding to the L-systems-based method are the best of 30 repeated optimization runs (the same applies later to Figure 6.21).



(a) Problem #1 (direct encoding) (b) Problem #2 (direct encoding) (c) Problem #6 (direct encoding)



(d) Problem #1 (SIMP) (e) Problem #2 (SIMP) (f) Problem #6 (SIMP)



(g) Problem #1 (L-systems) (h) Problem #2 (L-systems) (i) Problem #6 (L-systems)

Figure 6.19: Comparison of designs obtained for three representative problems #1, #2 and #6.

pb.	SIMP		direct encoding		L-systems		
	$\bar{T}$ [K]	$R_{\{\bar{T}\}} \times 10^3$	$\bar{T}$ [K]	$R_{\{\bar{T}\}} \times 10^3$	$\bar{T}$ [K]	$R_{\{\bar{T}\}} \times 10^3$	$\bar{Q} \times 10^{-3}$
#1	30.483	609.66	31.328	626.56	30.292 ± 0.012	605.84 ± 0.23	30.2
#2	17.934	358.69	18.888	377.75	17.844 ± 0.022	356.89 ± 0.44	33.9
#3	7.982	159.64	8.433	168.67	8.020 ± 0.035	160.41 ± 0.70	41.2
#4	4.635	92.70	4.955	99.11	4.680 ± 0.021	93.61 ± 0.42	41.6
#5	1.797	35.95	2.052	41.03	1.922 ± 0.028	38.44 ± 0.56	36.0
#6	0.397	7.95	0.500	9.99	0.459 ± 0.014	9.17 ± 0.27	54.7

Table 6.5: Results in numerical format, including the optimized average temperature $\bar{T}$ and the corresponding nondimensional thermal resistance $R_{\{\bar{T}\}}$, as well as the average number of required objective function evaluations, $\bar{Q}$. The results obtained by the L-systems-based method are reported as the mean and its 95% confidence interval. The estimate of required objective function evaluations of the direct encoding method is based on Section 2.4 and Figure 2 in Ref. [Boichot and Fan \(2016\)](#).

In optimization problem #2 (Subfigures b, e, h), all methods yield designs where the high-conductive material is distributed in patterns featuring only radial branches starting from the heat sink, with no bifurcations in the outmost regions of the domain. However, the numbers of radial branches in these designs are different, ranging from 6 to 12 – the design obtained by the L-systems-based method having the most branches.

In optimization problem #6 (Subfigures c, f, i), the design obtained by the L-systems-based method has a similar radial pattern of high-conductive material, whereas the corresponding designs with the direct encoding and SIMP methods have a bifurcating tree structure with three different scales. Despite having a fundamentally different topology, the average temperature of the design by the L-systems-based method (Subfigure i) is only 1.7% higher than the corresponding design reached by the SIMP method (Subfigure f).

Let us next examine the results for optimization problems #7-12, where the objective is to minimize the maximum temperature $T_{\max}$. Here, we only benchmark the L-systems-based method against the direct encoding method, as the SIMP method would require transforming the min-max problem into a new one involving the p -norm operator, which is continuous and differentiable (cf. the paper by [Yan et al. \(2018\)](#)). Making comparisons between both formulations would be problematic since they do not involve the same objective functions and because the solutions of the p -norm problem depend on the p value (that is usually selected based on different numerical tests). Figure 6.20 shows the benchmarking of the L-systems-based results against the direct encoding ones for optimization problems #7-12. The corresponding numerical data is presented in Table 6.6. As the results obtained are normally, or nearly-normally, distributed (see Figure E.2 in Appendix E), we again report the mean and 95% confidence interval (determined based on the standard error) of the results obtained using the L-systems-based method.

pb.	direct encoding			L-systems		
	$T_{\max}$ [K]	$R_{\{T_{\max}\}} \times 10^3$	$\bar{Q} \times 10^{-6}$	$T_{\max}$ [K]	$R_{\{T_{\max}\}} \times 10^3$	$\bar{Q} \times 10^{-3}$
#7	41.029	820.58	$\sim 5-7$	40.180 ± 0.093	803.59 ± 1.86	34.3
#8	29.380	587.61	$\sim 5-7$	27.665 ± 0.050	553.29 ± 1.00	29.5
#9	12.508	250.17	$\sim 5-7$	12.031 ± 0.105	240.61 ± 2.09	42.5
#10	6.740	134.80	$\sim 5-7$	6.499 ± 0.058	129.99 ± 1.17	38.8
#11	3.089	61.78	$\sim 5-7$	3.329 ± 0.101	66.58 ± 2.01	45.2
#12	0.833	16.66	$\sim 5-7$	0.967 ± 0.059	19.34 ± 1.18	60.3

Table 6.6: Results in numerical format (see the caption of Table 6.5 for explanations of the symbols).

In optimization problems #7-10, the L-systems-based method yields on average better results than the direct encoding. In optimization problems #11 and #12, the obtained designs are on average worse than those obtained by the direct encoding. These conclusions are also statistically significant.

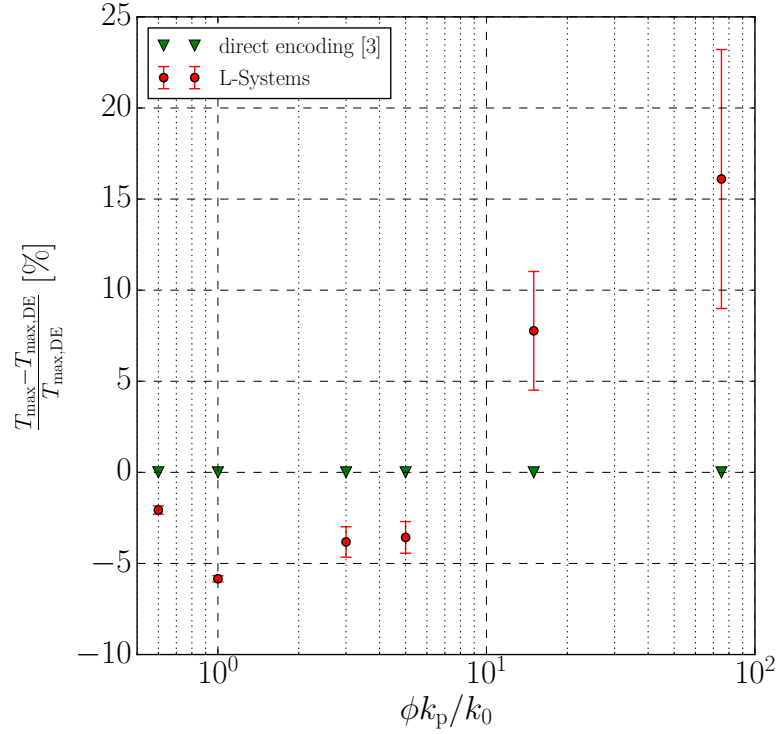


Figure 6.20: Benchmarking of the L-systems-based method against the direct encoding method (Boichot and Fan, 2016) for problems #7-12. $T_{\max,DE}$ is the maximum temperature optimized by the direct encoding method.

It is worth noticing that in optimization problems #9-12 the performance of the L-systems-based method gradually decreases against the direct encoding method as the dimensionless coefficients $\phi k_p / k_0$ increases. Thus, a crossover value, of around 5 to 15, may exist for the dimensionless coefficient $\phi k_p / k_0$, above which the direct encoding is, on average, more efficient than the L-systems-based method. However, such trend is here less clear than in the earlier results between the L-systems-based and the SIMP methods in Figure 6.18.

Nevertheless, also in optimization problems #11 and #12, the best designs obtained by the L-systems-based method are better than those obtained by the direct encoding method; the objective function values of these designs are 2.901 and 0.747 K, respectively.

The L-systems-based method requires significantly fewer objective function evaluations than the direct encoding method (Table 6.6) – the difference being of two orders of magnitude. However, we want to point out that the convergence criteria of the algorithms are different and the reporting of the number of required objective function evaluations in the reference study Boichot and Fan (2016) is limited. Nevertheless, even if the optimization problem #12 is the most demanding from a computational point of view, the entire set of 30 optimization runs requires $30 \times 60.3 \cdot 10^3 \approx 1.8 \cdot 10^6$ function evaluations, which is only around 36% of a single optimization run with the direct encoding. If we consider the set of optimization runs as a multi-start approach, the L-systems-based

method also yields a better result than the direct encoding method for problems #11 and #12.

We believe that there are two main reasons why the L-systems-based method outperforms the direct encoding method. First, the designs space of the method is channeled to favorable designs, in which the entire material distribution is fully connected and touches the heat sink. Second, as we mentioned in the introduction, L-systems (like other generative encodings) are construction recipes, which can be used to define diverse design spaces with relatively few design variables and are capable of producing designs consisting of self-similar and hierarchical components.

An example of self-similarity can be seen, for example, in the design in Figure 6.19(h). Considering either side of the symmetry axis, the material distribution of this design consists of two compositions of three radial spikes. These compositions are similar to each other, but of different scales.

Designs obtained by the L-systems-based and direct encoding methods are shown in Figure 6.21 for problems #7, #8 and #12. These problems have the same conductivity ratio k_p/k_0 and volume fraction ϕ as problems #1, #2 and #6, respectively, which results were presented earlier in Figure 6.19. The optimized designs for the other optimization problems (#9, #10 and #11) are presented in Figure F.2 in Appendix F.

In problems #8 and #12, the L-systems-based method also produces designs where most of the high conductive material is distributed in patterns having only radial branches (Figures 6.21(e) and 6.21(f)). However, these branches penetrate deeper in the finite-size volume and their tips are thicker than in Figures 6.19(h) and 6.19(i), mitigating high temperatures in the outmost regions of the domain, where the temperature increase is the most critical.

On the other hand, in problem #7, the obtained design with the L-systems-based method (Figure 6.21(d)) is significantly different to the corresponding design minimizing the average temperature $\bar{T}$ (Figure 6.19(g)). As the conductivity ratio k_p/k_0 is low, the critical regions for the maximum temperature are located at the two corners furthest away from the heat sink, which the method seeks to fill with high conductivity material.

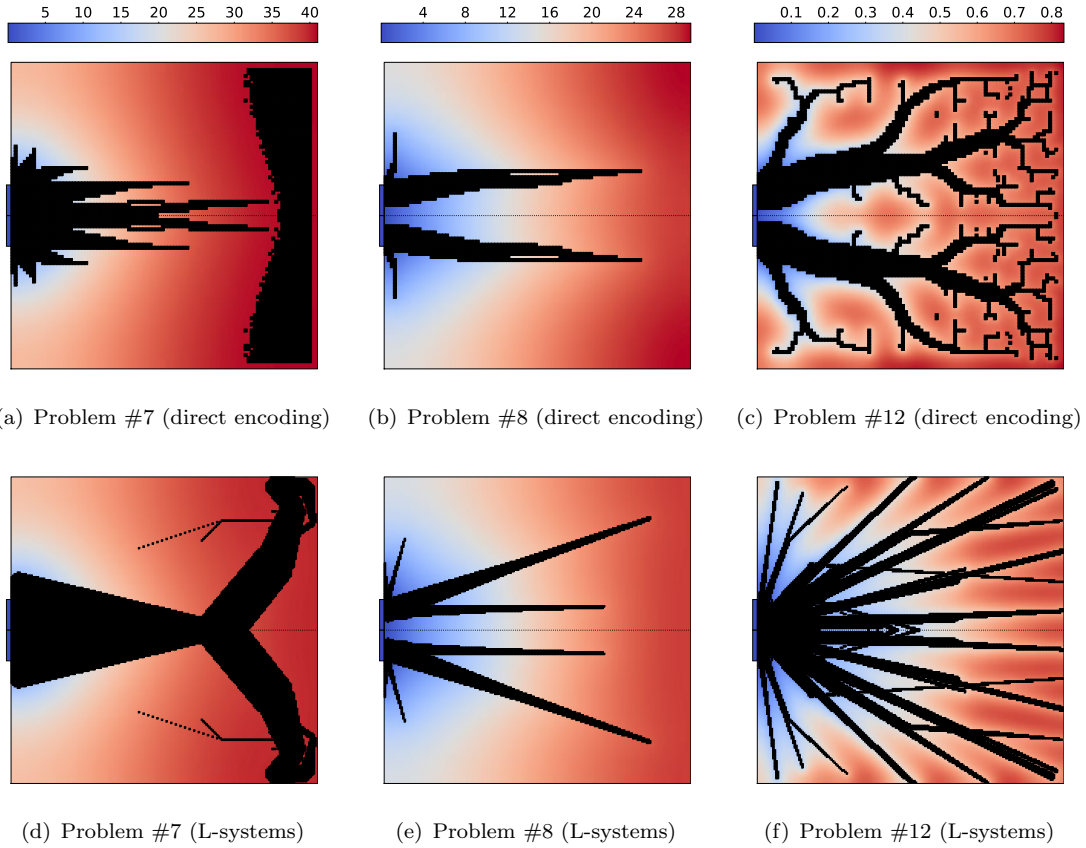


Figure 6.21: Comparison of designs obtained for three representative problems #7, #8 and #12.

6.3.7 Multi-objective optimization

The design of realistic engineering systems often quickly becomes multi-objective. Therefore, in this section, we demonstrate the suitability of the L-systems-based method to tackle multi-objective design optimization of heat conductors with both scalar and integer objectives. The purpose is to obtain a set of Pareto optimal designs in the objective space, which represent the best trade-offs between two competing objectives.

As the optimization algorithm, we here apply the NSGA-II (Section 2.3.4). We use the same control parameters as in the single-objective optimization with slight modifications. The tournament pool size is changed into two, as defined Deb et al. [Deb et al. \(2002\)](#). The implementation of NSGA-II in DEAP requires the population size to be a multiple of four ([Fortin et al., 2012](#)), so we change it to be 152. Finally, we terminate an optimization after 300 generations.

Let us first examine an optimization run where the objectives are to concurrently minimize the average and maximum temperatures, $\bar{T}$ and $T_{\max}$, which were individually minimized in the previous section. We define the conductivity ratio k_p/k_0 and volume

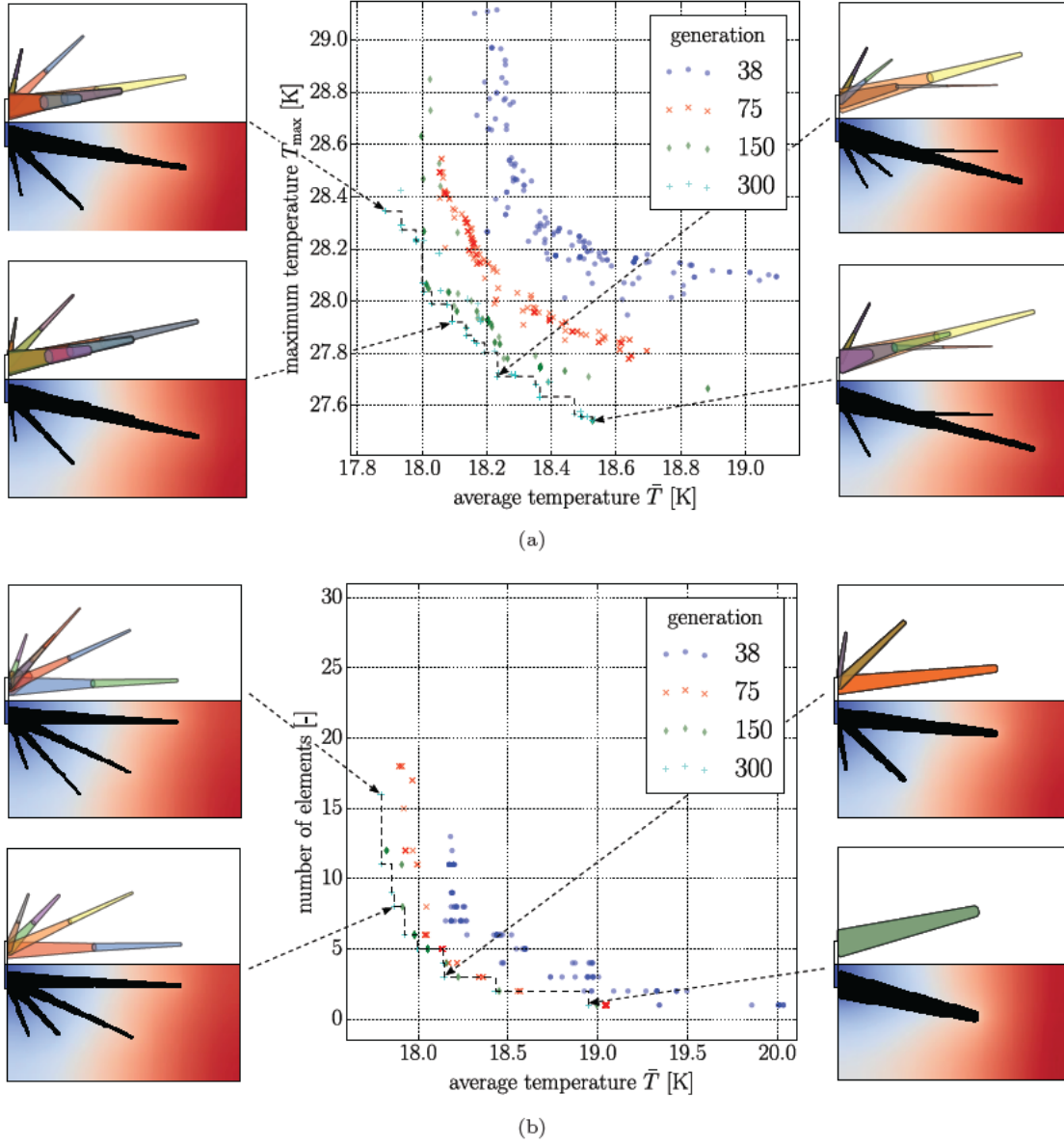


Figure 6.22: Results of two multi-objective optimization runs, evolved via NSGA-II. The figures on the sides show representative designs from the Pareto front, indicated by the dashed line.

fraction ϕ to be the same as in the optimization problems #2 and #7. Figure 6.22(a) illustrates the bi-objective optimization process of approaching the Pareto front, as well as representative designs lying at the approximated Pareto front. The designs lying at the ends of the approximated Pareto front compare well with corresponding single-objective results. Thus, the entire Pareto front represents designs that could be considered by a chip manufacturer who would like to reach an average temperature as low as possible, while reducing the hot spots over the component.

Finally, let us dive deeper into the mindset of the hypothetical chip manufacturer. After seeing some of the designs in Figures 6.19 and 6.21, he or she might question whether

there is any compromise design that provides a good heat transfer with lesser geometrical complexity. The L-systems-based method provides one possible measure for ‘design complexity’: the number of steps taken by the turtle. Let us refer to this measure as the number of elements. Figure 6.22(b) shows a bi-objective version of solving the optimization problem #2, using the number of elements as the second objective. Clearly, the resulting designs can be chosen to be much simpler in shape, albeit at the expense of conductive performance.

6.4 Conclusions

In this chapter, we applied our versions of the ground structure approach and the parametric L-system-based method to the conductive topology optimization problem defined by Bejan (1997). We found that, when using the ground structure approach, the optimized average temperature $\bar{T}$ is highly dependent on the chosen ground structure type and density.

The L-systems-based method, the implementation and evaluation of which were the main objective of this chapter (see the reasoning in Section 2.9), yielded, on average, 24.1% lower optimized average temperatures than the ground structure approach on the initial optimization problem (which parameters were listed in Table 6.2).

Next, we benchmarked the L-systems-based method against two relevant topology optimization methods in the literature, i.e. the GA-based direct encoding (Boichot and Fan, 2016) and the SIMP method (Marck et al., 2012), on several different optimization problems (which parameters were listed in Table 6.4). We obtained statistical significance that the L-systems-based method yields better designs than the direct encoding in 10 out of 12 tested optimization problems. Further, our results indicate that the method yields lower objective function values than the widely used and well established SIMP method in optimization problems, the dimensionless coefficient $\phi k_p/k_0$ of which is less or equal to 1.

One of the motivations to conduct this work has been to find improvements to evolutionary topology optimization, which have not gained significant acceptance in the topology optimization community. Evolutionary topology optimization methods are often criticized because of their difficulty of ensuring structural continuity in the designs (Munk et al., 2015) and because they require orders of magnitude more function evaluations than the gradient-based topology optimization methods (Sigmund, 2011) (see Chapter 1). The (evolutionary) L-systems-based method, we use in this chapter, naturally produces designs with full structural connectivity and requires two orders of magnitude fewer function evaluations than the direct encoding method.

In addition, we demonstrated the suitability of the L-system-based method to multi-objective optimization of an electrical device cooling system, involving both temperature and manufacturing related objectives. As the method does not rely on the gradient information, it is (unlike the SIMP method) also applicable to integer, or other non-differentiable, objective functions.

Despite having the flexibility of generating branching tree-structures, the L-systems-based method yielded mostly designs, in which the dominant feature is the radial pattern of high conductive material (see Figures 6.19(h), 6.19(i), 6.21(e) and 6.21(f)). This observation is in line with the conclusions of the recent study by Yan et al. (2018). The authors initiated the SIMP method from rank-1 laminates, which resulted in radial material distributions they refer to as *lamellar needle structures*. The authors showed that these structures have lower average and maximum temperatures than branching tree-structures, typically considered as the optimal structural type in the literature, in several test cases.

The interpretation of L-systems into three-dimensional geometries is already an established method in computer graphics to represent biological organisms (Prusinkiewicz and Lindenmayer, 2012). Thus, the methodology presented here is extensible to three-dimensional topology optimization, simply by adding two new symbols to the alphabet Σ and the corresponding numerical representation. These symbols command the turtle to pitch up or down or roll with respect to its previous heading.

The method is also applicable to fields outside thermal systems, such as urban planning or designing escape routes in music festival areas, airports or large sports arenas (Gersborg-Hansen et al., 2006), in other words, to applications that involve volume-to-point or area-to-point problems.

Chapter 7

Application III: Integrally stiffened panel

The main objective of this chapter is to parameterize the stiffener layout of an integrally stiffened panel using map L-systems, and apply the parameterization method to the fundamental natural frequency maximization of the structure.

In Section 2.8.3, we listed the following advantages of the map L-systems-based method for the purpose. First, when using the method, the phenotypes can be ensured to consist of stiffener-like structural members, facilitating the manufacturing of the final design. Second, unlike the turtle interpretation of L-systems, the map L-systems yield phenotypes that do not have ‘dead end’ stiffeners. Presumably, structural discontinuities would not be beneficial for the current application. Third, as we have already mentioned in the context of the previous applications, the method does not require a priori definition of candidate structural members.

Throughout this work, we have investigated whether direct or generative encodings are more suitable for the studied optimization problems. Therefore, also here, we apply the ground structure approach (i.e. a direct encoding method) to the optimization problem. Finally, we benchmark the results against integrally stiffened panels with commonly used iso- and orthogrid layouts, with optimized stiffener spacings.

7.1 Optimization problem

The objective of the optimization problem is to maximize the fundamental natural frequency ω_f of an integrally stiffened aluminum panel, subject to mass and manufacturing

Property	Value	Unit
panel dimensions $l \times l$	1.0×1.0	m
total mass m_{panel}	8.0	kg
panel thickness t_{skin}	1.0	mm
stiffener aspect ratio $h_{\text{stiffener}}/t_{\text{stiffener}}$	7.5	-
Young's modulus E	73.1 ⁱ	GPa
Poisson's ratio ν	0.33 ⁱ	-
density ρ	2780 ⁱ	kg/m ³

ⁱ www.aerospacemetals.com (accessed on 2nd August 2017)

Table 7.1: Properties of the optimization problem.

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 7.1 for geometrical and material properties).

The material is intended to be removed via face milling or chemical etching¹. We impose two manufacturing constraints: the minimum wall thickness is 1 mm, and all stiffeners (in a design) 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.

We evaluate the fundamental natural frequency ω_f of individuals by finite element(FE)-based modal analysis. The analyses 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 the four edges of the panel. Both the skin panel and the stiffeners are modeled using first order shell elements. The skin sections between the stiffeners are meshed using a quad-dominated algorithm, producing both triangular and quadrilateral elements. The geometry of all stiffeners is a rectangle, and therefore they are meshed using only computationally more efficient quadrilateral elements. The mesh density of the FE model will be chosen in Section 7.4.

¹The manufacturing constraints, we here impose, also enable the manufacturing technique where the stiffeners are attached to the panel using friction stir welding.

7.2 Methods

This section describes in detail the implementation of the ground structure approach and the map L-systems-based method to the current topology optimization problem. We evolve both parameterizations by genetic algorithms (GAs).

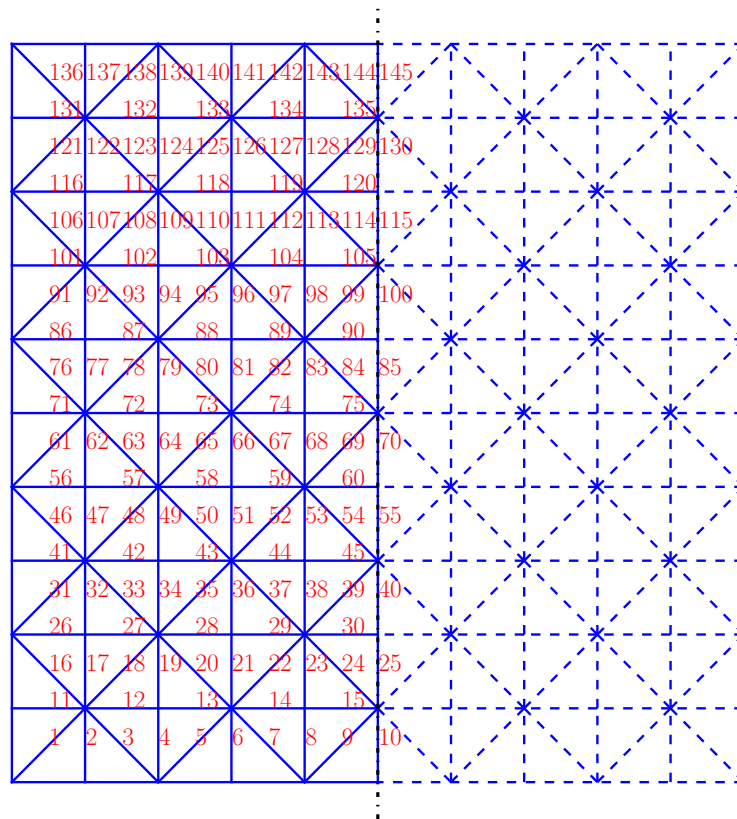
7.2.1 Ground structure approach

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 symmetrical at least with respect to one of these axes. Thus, we define the ground structure inside a rectangular domain that represents half of the stiffener layout on the plate structure (see the continuous lines in Figure 7.1(a)). The other half of the stiffener layout is generated by mirroring the first with respect to the vertical symmetry axis (see the dashed lines in the same figure). As the edges of the plate are defined to have pinned boundary conditions, we omit the stiffeners lying on them. Figure 7.1(b) shows the full ground structure mirrored and mapped to the panel structure.

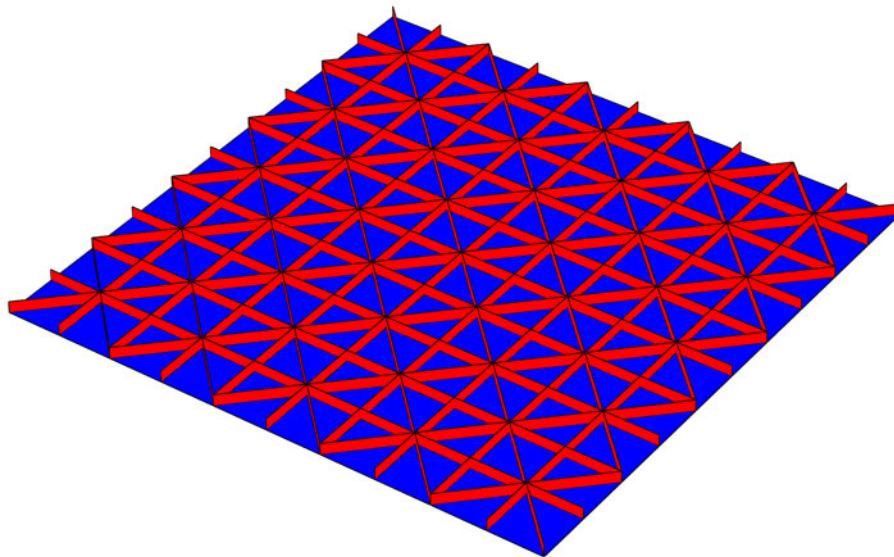
We again use the same types of ground structure as in Chapters 3 and 6, i.e. the quadrilateral with diagonals (Figure 7.1) and hexagonal ground structure (Figure 7.2). Moreover, we vary the density of the ground structure using the density measure $m \times n$, defined in Section 6.2.1². Both ground structures visualized in Figures 7.1 and 7.2 have a density of 10×5 .

We encode the existence of ground structure members into the genotype $\mathbf{x}$ as binary variables. Figure 7.1 shows the order of these variables in the case of the quadrilateral ground structure.

²However, as the orientation of the ground structure here is different, m and n are the numbers of boundary segments in the west and south boundaries, respectively.



(a) mirroring and indexing of the ground structure members



(b) full ground structure

Figure 7.1: Rectangular ground structure with diagonals having a density of 10×5 .

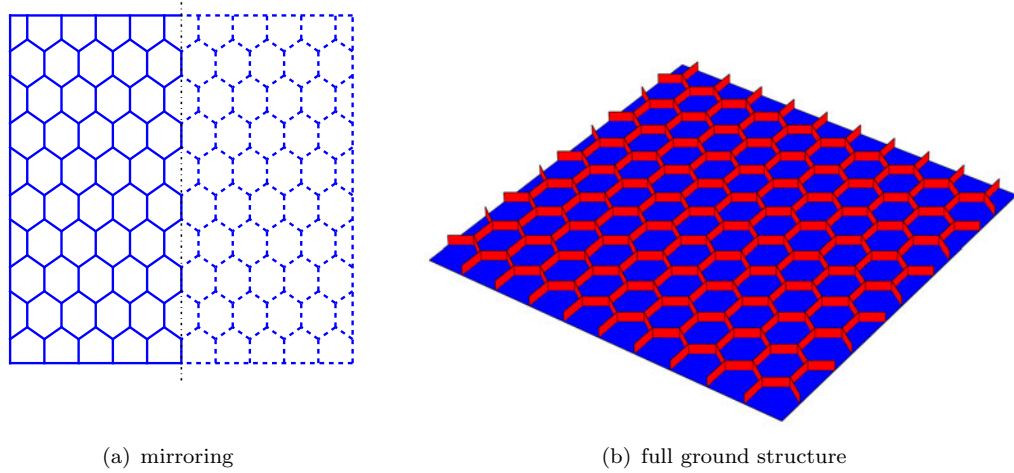


Figure 7.2: Hexagonal ground structure having a density of 10×5 .

7.2.2 Map L-systems-based method

Earlier in this work, we have described how the phenotypes of map L-systems are constructed based on the axiom, rewriting rules and additional variables (Section 2.7.2) and how we encode the axiom, rewriting rules and additional variables into the numerical format suitable for GAs (Sections 2.7.3 and 4.1). We here use the map L-systems to represent the stiffener layout on the panel structure.

We assume, in the same way as with the ground structure approach, that the optimal stiffener topology is symmetric at least with respect to one of the symmetry axes. Thus, we define the axiom to be a rectangle, covering half of the plate domain (see nodes 1-4 and the continuous lines in Figure 7.3(a)), and define the second half to be the mirror image of the first half.

We represent the existence of the edge $\{4, 1\}$, laying on the symmetry axis, by an additional, boolean design variable. The boundary edges $\{1, 2\}$, $\{2, 3\}$, $\{3, 4\}$ and their mirrored counterparts are excluded from the stiffener topology.

Figure 7.3 demonstrates the mirroring and mapping of a map L-system phenotype into the panel structure. The phenotype is the third developmental stage of the map L-system described in Equation 2.24.

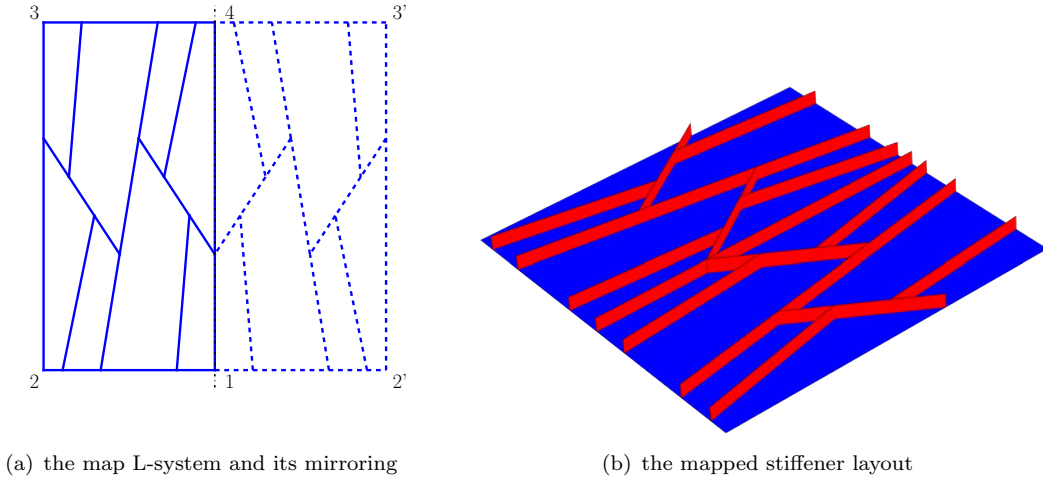


Figure 7.3: The third developmental stage of the map L-system in Equation 2.24 mapped as a stiffener layout.

7.2.3 Scaling

Once the stiffener layout is generated, we scale the cross-sectional area A_{cs} of the stiffeners³ to satisfy the mass constrained, using equation

$$A_{cs} = \frac{m_{\text{stiffener}}}{\rho l_{\text{tot}}}, \quad (7.1)$$

where $m_{\text{stiffener}}$ is the mass budget of the stiffeners and l_{tot} is the total length of the stiffeners in the panel structure. The mass budget of the stiffeners, $m_{\text{stiffener}}$, is equal to the total mass constraint m_{panel} subtracted by the panel mass $\rho l^2 t_{\text{skin}}$.

Using the defined stiffener aspect ratio, the stiffener thickness $t_{\text{stiffener}}$ and height $h_{\text{stiffener}}$ are

$$\begin{cases} t_{\text{stiffener}} = \sqrt{\frac{A_{cs}}{7.5}} \\ h_{\text{stiffener}} = \sqrt{7.5 A_{cs}}. \end{cases} \quad (7.2)$$

The stiffener height $h_{\text{stiffener}}$ is modeled as the actual height of the stiffener in the FE model, and the stiffener thickness $t_{\text{stiffener}}$ as a parameter of the corresponding shell elements.

7.3 Reference designs

First, we need to determine the optimal stiffener densities for reference designs with iso- and orthogrids, which maximize the objective function ω_f . We use the number of

³In the definition of the optimization problem, the stiffeners were constraint to have the same size (see Section 7.1).

stiffeners in the x -direction (see the coordinates in Figure 7.4) of the panel as a design variable. With orthogrids, the number of stiffeners in the y -direction is the same as in the 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 Figure 7.4, where fundamental mode shapes of representative designs are shown. The maximum fundamental natural frequency ($\omega_f = 143.39$ Hz) corresponds to the orthogrid design with four 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.

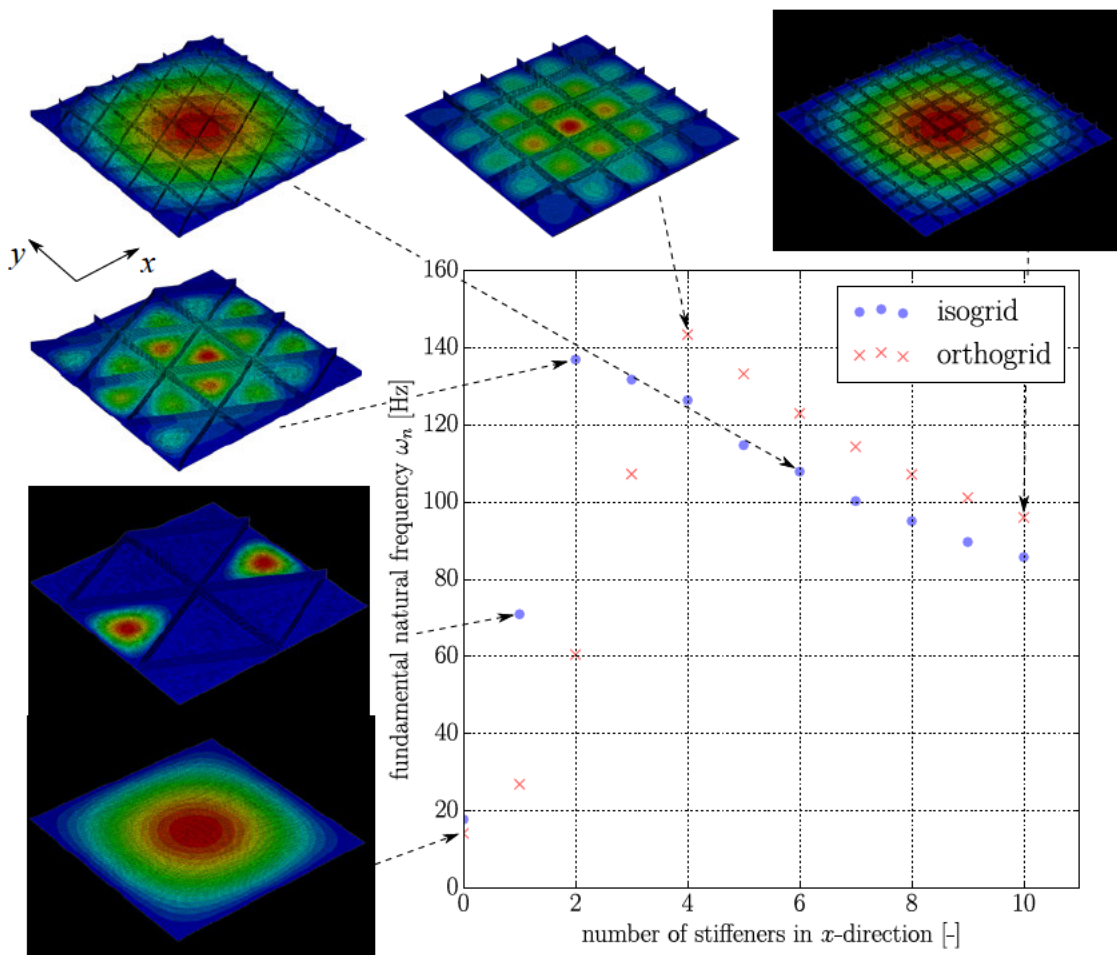


Figure 7.4: Reference designs using iso- and orthogrids with variable stiffener densities. The orthogrid with four stiffeners in both directions has the highest fundamental natural frequency of $\omega_f = 143.39$ Hz. The contour plots illustrate fundamental mode shapes of representative designs.

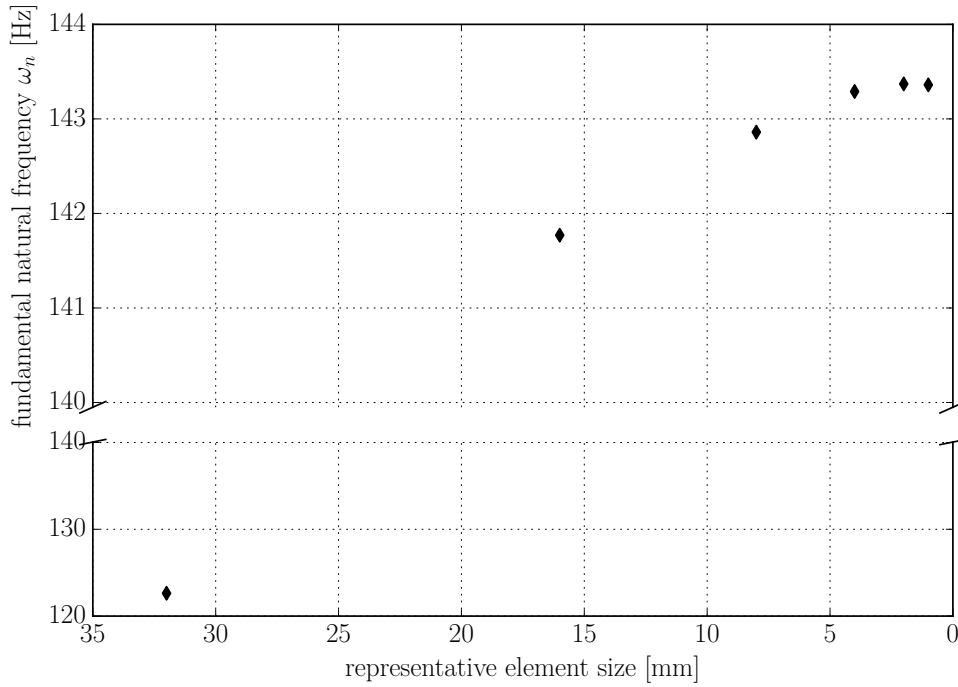


Figure 7.5: Mesh density verification of the baseline design, i.e. the orthogrid with five seeds a side.

7.4 Mesh density verification

As described already in Application I (Section 3.7.4.1), FE analysis results are dependent on the mesh density. Selecting the mesh density is a trade-off between accurate results and high computational cost. In this section, we examine an adequate mesh density for the current application. We use the baseline design as a representative structure for the mesh density study.

We tested the model using six mesh densities; Figure 7.5 shows the fundamental natural frequency as a function of representative element size. Typically, coarse mesh stiffens the structure, and leads to the overestimation of natural frequencies. However, in our application, the coarser the mesh density is, the more the fundamental natural frequency is underestimated. As the behavior is rather unexpected, we also studied the effect of mesh density on a structure containing only the skin panel (orthogonal design with no stiffeners in the x -direction in Figure 7.4). With this structure, we observed the typical behavior: the coarser the mesh is, the more the fundamental natural frequency is overestimated. The reason for the opposite behavior with the baseline design is unknown, but could be related to a coarse mesh not being able to represent the fundamental mode shape, which would also affect the mass matrix of the FE model. A similar opposite dependency can be seen, for example, in the mesh density study by Liu and Glass (2013).

For the remaining of this chapter, we choose to use the representative element size of 8 mm for the skin plate and stiffeners in their longitudinal direction. We found the fundamental natural frequency with this element size to be 0.35% smaller than that with the smallest tested element size. During an optimization process, some designs may have significantly smaller stiffener height. To ensure an adequate number of elements also in the shortest direction of these stiffeners, we halve the representative element size to 4 mm in their transverse direction⁴.

7.5 Results

In this section, we, first, examine the results obtained by the ground structure approach and map L-systems-based method separately and, finally, benchmark them against the baseline design, presented in Section 7.3.

7.5.1 Ground structure type and density

Here we evolve the ground structure approach parameterization, defined in Section 7.2.1, using a single-objective GA⁵. We choose to use the modified control parameter combination #4, because we found it to yield the best final designs (out of the three studied parameter combinations) in the previous application (see Section 6.3.1). We acknowledge that most likely a better control parameter combination exists for the current application – nevertheless, this combination is our most educated guess based on the earlier experiments of this work.

Earlier in this work, we observed that the results obtained by the ground structure approach are highly dependent on a priori choices of the ground structure type and density. The main purpose of this section is to study the effects of these two on the results of the current optimization problem. We conducted a set of experiments with both the quadrilateral and hexagonal ground structures of densities 6×3 , 8×4 , 10×5 and 12×6 . Optimization runs with each ground structure were repeated five times.

Figure 7.6 presents the obtained distributions of the optimized fundamental natural frequency ω_f and the number of objective function evaluations Q . The average number of required objective function evaluations, $\bar{Q}$, seems to increase with the density of the ground structure. However, the deviation of the results is large, especially with quadrilateral ground structures of densities 8×4 , 10×5 and 12×6 .

The densities that yield, on average, the highest optimized fundamental natural frequencies with the quadrilateral and hexagonal ground structures are 8×4 and 6×3 ,

⁴The presented results for the reference designs (Figure 7.4) were generated using the representative element sizes specified herein.

⁵See Section 2.3 for a review of GAs.

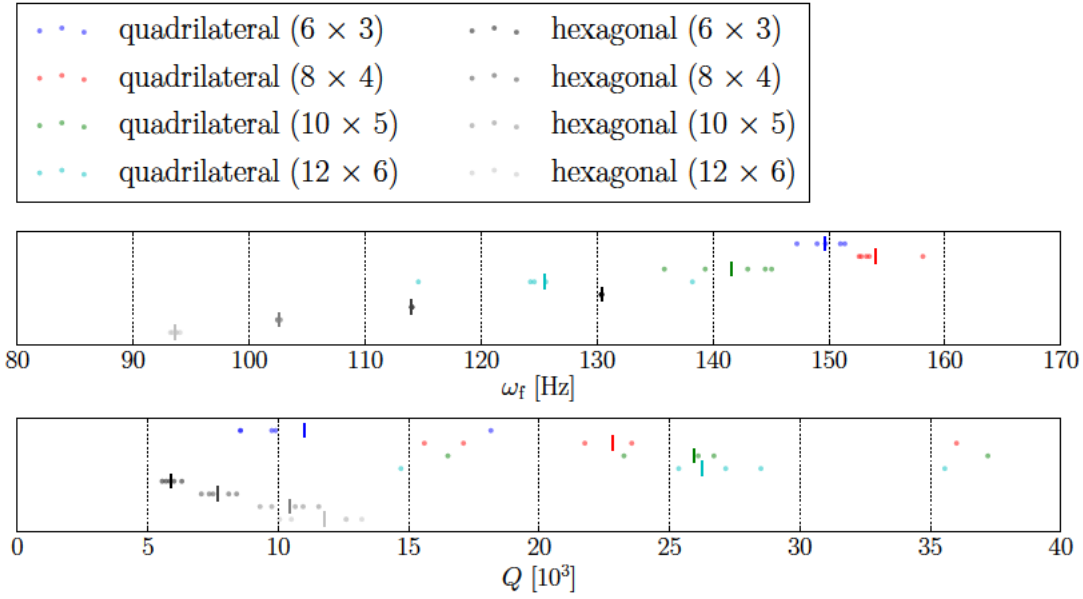


Figure 7.6: Distributions of the optimized fundamental natural frequency ω_f and the number of objective function evaluations Q from optimization runs with various ground structure types and densities. The vertical lines indicate the average values of the quantities.

respectively. These ground structures represent the second coarsest and the coarsest of the tested ground structure densities, respectively. Starting from these densities, the finer the ground structures are, the worse designs, on average, are obtained. Let us compare the quadrilateral ground structures of densities 6×3 (the coarsest) and 12×6 (the finest). The design space of the latter includes all the designs of the former, but designs obtained with it have, on average, 16.2% lower fundamental natural frequencies than with the former. The result indicates that the design space of the finest ground structures is too large for the GA to find good designs. Earlier in Section 6.3.2, we obtained a similar result in the heat transfer application.

However, in the current optimization problem, we believe that this is only the case for the quadrilateral ground structures. To explain our reasoning, let us look at the best designs obtained with hexagonal ground structures in Figure 7.7. These designs consist of all, or nearly all, ground structure members. The non-existing ground structure members, if any, are located near the corners of the plate structure. Here, we draw a conclusion that, for this optimization problem, the hexagonal layout of stiffeners is only competitive if it is complete – any missing structural member in the layout is a potential structural weakness. Thus, the global optimum in these design spaces might not be much better than those shown in Figure 7.7.

Following this conclusion, we conducted a search of the optimal density of the complete hexagonal stiffener layout (in similar fashion as we studied the densities of iso- and orthogrids in Section 7.3). Figure 7.8 shows the fundamental natural frequency ω_f as

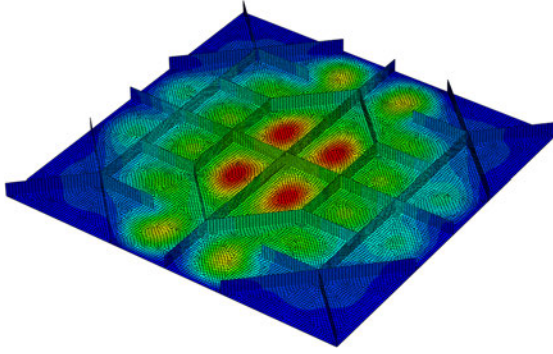
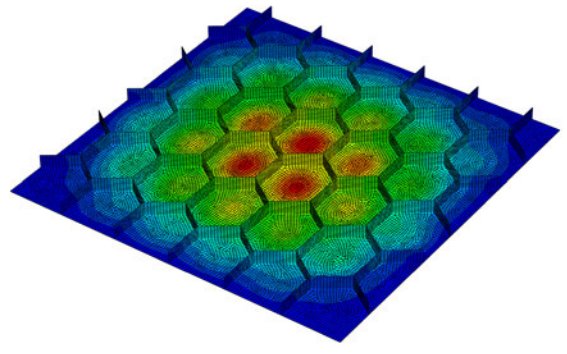
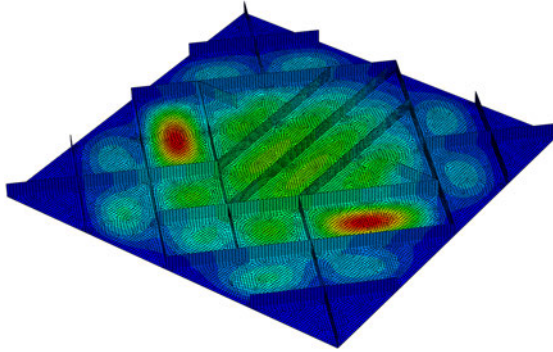
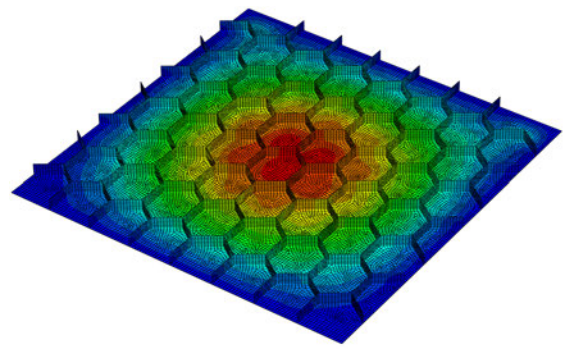
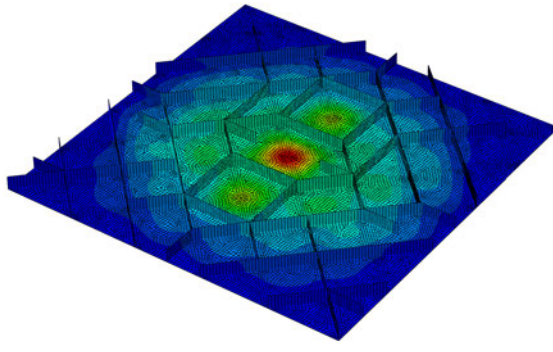
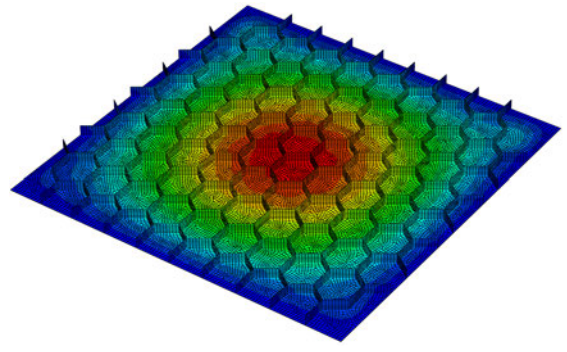
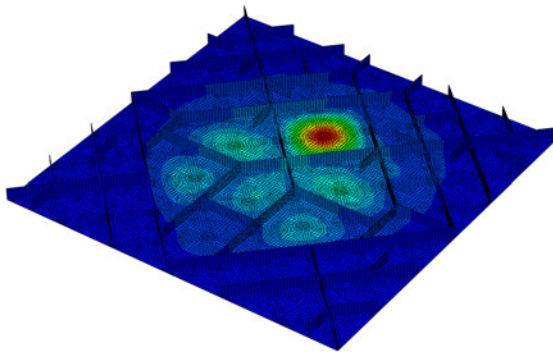
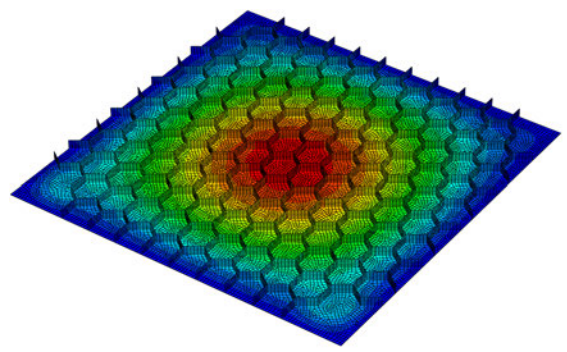
(a) Quadrilateral (6×3) ($\omega_f = 150.98$ Hz)(b) Hexagonal (6×3) ($\omega_f = 130.37$ Hz)(c) Quadrilateral (8×4) ($\omega_f = 158.10$ Hz)(d) Hexagonal (8×4) ($\omega_f = 114.02$ Hz)(e) Quadrilateral (10×5) ($\omega_f = 145.05$ Hz)(f) Hexagonal (10×5) ($\omega_f = 102.66$ Hz)(g) Quadrilateral (12×6) ($\omega_f = 138.22$ Hz)(h) Hexagonal (12×6) ($\omega_f = 94.02$ Hz)

Figure 7.7: The best obtained designs using the ground structure approach. The contours indicate the fundamental mode shape.

the function of boundary segments in x -direction, ranging from 1 to 12. The optimal structure is obtained by five boundary segments and has the fundamental natural frequency of $\omega_f = 137.61$ Hz. The value is higher than that of optimal density isogrid panel ($\omega_f = 136.93$ Hz), but lower than that of the corresponding orthogrid panel ($\omega_f = 143.39$ Hz).

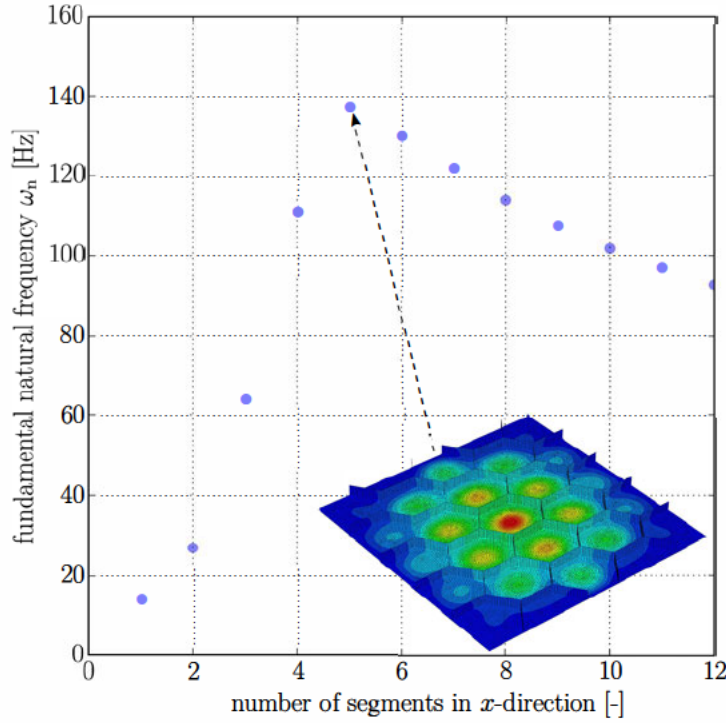


Figure 7.8: Hexagonal stiffener layout with a variable stiffener density. The layout with the highest fundamental natural frequency has five cells in the x -direction ($\omega_f = 137.61$ Hz).

Finally, let us examine the best designs obtained using the quadrilateral ground structures with diagonal members (Figure 7.7). The main characteristic of these designs is the presence of one to three circumferential stages of diagonal ground structure members. In addition, the designs have radial lines of stiffeners that initiate from the four corners of the panel⁶. Three (Figures 7.7(a), 7.7(e) and 7.7(g)) out of four of these designs are symmetric, or nearly-symmetric, with respect to the horizontal symmetry axis.

7.5.2 Control parameters in the map L-systems-based method

In Chapter 4, we showed a systematic process for identifying the optimal parameter combination for a particular problem with the map L-systems-based parameterization and we demonstrated this across a set of ‘lightweight’ test cases. We obtained the

⁶An exception is the design obtained with the ground structure having a density of 10×5 , in which only two of the radial lines initiate from the corners.

Pareto front in the space of fitness of the final solution and the average number of required objective function evaluations. We also put forward the correlation analysis presented there, showing strong correlation across the test cases, and thus indicating that the results have only a weak problem-dependency. As the statistical experiment was conducted using the 'lightweight test cases, we study here whether we can see results with similar trends in the current optimization problem, which represent a 'real-life' engineering application⁷.

Ideally, we would conduct a similar statistical experiment 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' problems.

We repeated the optimization process 10 times with control parameter combinations #1, #4, #7 and #12 (Table 4.4). These combinations represent the 'practical' Pareto front (see the reasoning in Section 4.5). Figure 7.9 shows the distributions of the optimized fundamental natural frequency ω_f and the number of objective function evaluations Q .

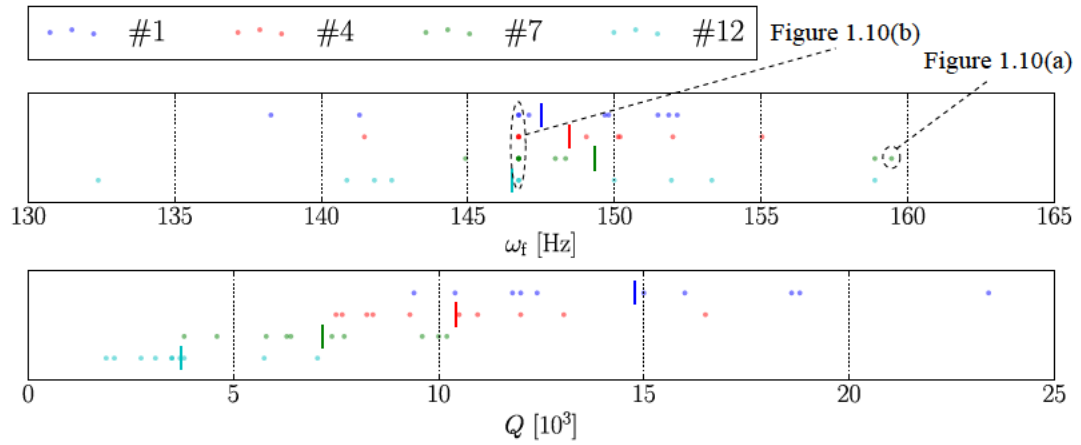


Figure 7.9: Distributions of the optimized fundamental natural frequency ω_f and the number of objective function evaluations Q from optimization runs using the map L-systems-based method with control parameter combinations #1, #4, #7 and #12. The vertical lines indicate the average values of the quantities.

Based on the results, the tested control parameter combinations #1, #4, #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. On average, the worst designs are obtained with the control parameter combination #12, which is in line with the statistical experiment, and the best designs are with the parameter combination #7, which is in contrast to the statistical experiment. However, as the standard deviation of the optimized fundamental natural frequency ω_f is large (3.69 to 7.54 Hz) and the

⁷We conducted a similar trade-off study with an L-systems-based parameterization in Section 6.3.5.1. However, in that study, we used the turtle interpretation (instead of map L-systems) as the interpretation formalism of L-systems.

sample size is relatively small, we cannot draw any statistically significant conclusions from the data⁸.

The best design (Figure 7.10(a)), which convergence history is shown in Figure 7.11, was obtained using the parameter combination #7. It consists of two radial lines of stiffeners, 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

$$\begin{aligned}
 \text{Axiom:} & \quad \omega_0 = CCAC \\
 \text{Rules:} & \quad P_1 : A \rightarrow [+ \vec{D}][+B]C[+ \overleftarrow{D}]A \\
 & \quad P_2 : B \rightarrow [-B][+D][-D]C[- \vec{D}][+ \vec{B}] \\
 & \quad P_3 : C \rightarrow [+D]B[-D]B[- \overleftarrow{C}] \\
 & \quad P_4 : D \rightarrow [- \vec{B}]DB[+C]CD \\
 \text{Additional variables:} & \quad f_a = 0.43263 \\
 & \quad n = 3
 \end{aligned} \tag{7.3}$$

Figure 7.10(b) shows a particular design that occurs frequently among the optimized designs (13 out of 30 optimization runs yield this design). The design consists of one stiffener lying at the vertical symmetry axis and seven evenly spaced stiffeners lying perpendicular to the first. The design is symmetric with respect to both vertical and horizontal symmetry axes.

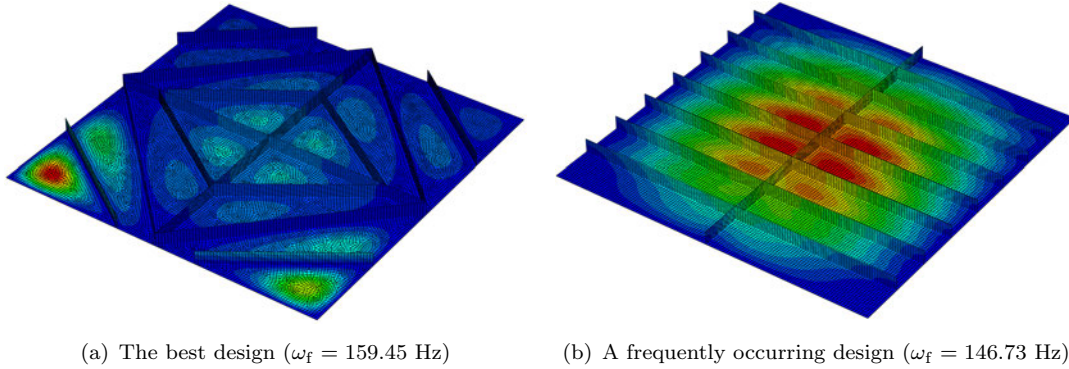


Figure 7.10: Representative designs obtained using the map L-systems-based method.

⁸Using the two-tailed Mann-Whitney U-test, the smallest p -value of the pairwise tests was $p = 0.440$ (between parameter combinations #7 and #12).

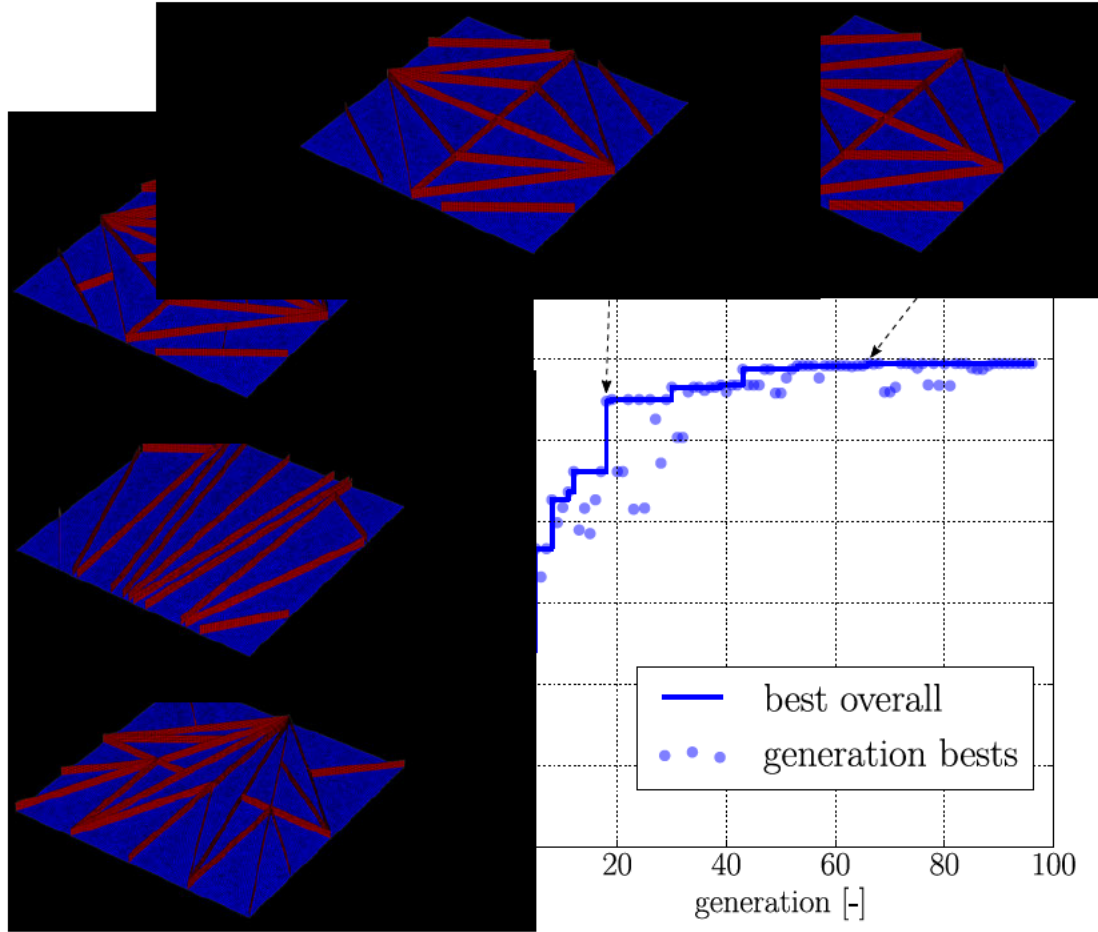


Figure 7.11: The convergence history and representative designs of the optimization run yielding the best design in Figure 7.10(a).

7.5.3 Benchmarking

Let us now benchmark the topology optimization results against the reference designs. Here, we pick the data of those repeated runs on the ground structure approach and the map L-systems-based method that yield, on average, the best designs. The former corresponds to the quadrilateral ground structure of density 8×4 and the latter to the control parameter combination #7.

The average and the highest fundamental natural frequency ω_f , and the number of objective function evaluations, Q , of these data sets are listed in Table 7.2. The table also shows the corresponding values for the reference designs, as well as relative differences with respect to the baseline design, i.e. the orthogrid with four stiffeners in x -direction.

Both topology optimization methods yield, on average, better designs than the baseline design. In fact, all five designs obtained by the ground structure approach and nine out of 10 designs obtained by the map L-systems-based method are better than the baseline design. Further, the ground structure approach yields, on average, better designs than

Method	ω_f [Hz]	relative difference [%]	$\bar{Q}$ [10^3]
isogrid layout	136.93	-4.51	-
orthogrid layout (baseline)	143.39	0.0	-
hexagonal layout	137.61	-4.03	-
ground structure approach ⁱ			
average ⁱⁱ	154.03	7.42	22.80
highest	158.10	10.26	-
map L-systems ⁱⁱⁱ			
average ^{iv}	149.32	4.14	7.18
highest	159.45	11.20	-

ⁱ Quadrilateral ground structure with diagonal members (density 8×4) ⁱⁱ repeated 5 times ⁱⁱⁱ Control parameter combination #7 ^{iv} repeated 10 times

Table 7.2: Benchmarking of topology optimization results against the baseline design, i.e. the orthogrid with four stiffeners in x -direction. ω_f and Q are the optimized, fundamental natural frequency and the number of required objective function evaluations, respectively.

the map L-systems-based method. However, the map L-systems-based method yields the best known design, which fundamental natural frequency is higher than the baseline design by a margin of 11.2%.

The map L-systems-based method requires, on average, around one third of the objective function evaluations of the ground structure approach. However, it should be noted that the data set corresponding to the map L-systems-based method is generated with the population size of $N_{\text{pop}} = 100$, while the corresponding value for the ground structure approach is $N_{\text{pop}} = 150$. Nevertheless, the average number of objective function evaluations of the data set generated with the control parameter combination #4 ($N_{\text{pop}} = 150$) is also less than half ($\bar{Q} = 10.41 \cdot 10^3$) of that of the ground structure approach.

7.6 Conclusions

In this chapter, we applied the ground structure approach and the map L-systems-based method to the design optimization of an integrally stiffened aluminum panel, manufactured via face milling. The ground structure approach yielded, on average, slightly better designs than the map L-systems-based method. However, we observed that the optimized designs by the ground structure approach are highly dependent on the predefined ground structure, and the method requires, on average, more than twice the amount of objective function evaluations in comparison to the map L-systems-based method.

Interestingly, the optimization runs with the hexagonal ground structures yielded optimized designs in which all, or nearly-all, of its members were existing. Any non-existing

ground structure member in the regions of the panel other than its corners seems to introduce a structural weakness. This indicates that, in this optimization problem, the hexagonal ground structure is not suitable for the ground structure approach, but should be considered as a complete layout. We sought the optimal density for the complete hexagonal layout – the resulting structure has a fundamental natural frequency lower than the baseline design by a margin of 4.03%.

The best designs obtained by the ground structure approach and the map L-systems-based method have similar features (see Figures 7.7(c) and 7.10(a)); the common characteristics are the circumferential stages and radial lines⁹ of stiffeners. These designs have fundamental natural frequencies higher than the baseline design (i.e. the orthogrid with four stiffeners in x -direction) by the margins of 10.26% and 11.20%, respectively. This raises an interesting question whether such layouts, e.g. the one shown in Figure 7.10(a), could be used more extensively in vibration critical engineering applications, instead of the traditional iso- and orthogrid layouts. However, here we emphasize that, due to the high computational cost of the study, our conclusion is only based on one optimization problem setup. Future research should include variable dimensions and mass constraints in the optimization problem to verify the hypothesis.

⁹In the designs obtained by the ground structure approach, the radial lines of stiffeners lie on the diagonals of the panel, whereas, in the designs obtained by the map L-systems-based method, they lie on the vertical and horizontal symmetry axes.

Chapter 8

Conclusions

In this work, we studied evolutionary topology optimization methods with direct and generative encodings. We narrowed these encodings to the ground structure approach and to those based on L-systems, respectively. In the next two sections, we present the contributions of this work, which correspond to the hypotheses defined in Section 2.9. The contributions are categorized in terms of the two research aims of this work.

8.1 Improving the performance

Motivated by the benefits of evolutionary algorithms, listed in the introduction (Chapter 1), the first research aim of the work was to improve the performance, i.e. the effectiveness and efficiency¹, of the evolutionary topology optimization methods. In the literature, the poor efficiency of the methods has been indicated as one of the major drawbacks, limiting their wider acceptance in the topology optimization community (Munk et al., 2015, Sigmund, 2011).

Contribution 1: In the literature, two-dimensional encoding is indicated to be particularly suitable for optimization problems with two-dimensional architectures, due to their better geographical-linkage between genotype elements in comparison to one-dimensional encoding. However, we obtained statistical significance that one-dimensional encoding, with the two-point crossover, is both more effective and more efficient than two-dimensional encoding on the application of sUAV wing internal structure. This result is against the hypothesis. Further, we obtained statistical significance that the distributed crossover is more effective than the two-point crossover (on the application in question). Later, in Chapter 6, we obtained the same result on the application of the heat conductor.

¹In Section 1.1, we defined the performance of an algorithm to be evaluated based on two quantities: its effectiveness (the ability of finding good solutions) and efficiency (the ability of finding them quickly).

Contribution 2: We conducted a statistical experiment of the effects of the genetic control parameter choices on the performance of the map L-systems-based method (Chapter 4). The experiment involved 432 control parameter combinations, which we evaluated on five low-cost test cases. Depending on the parameter combination, the average completion rate of the resulting algorithm on finding the global optimum ranged from 0.0 to around 0.7.² This result supports the hypothesis that the control parameter choices have a significant effect on the performance of the algorithm. We reported the best performing combinations as a Pareto front in the space of effectiveness and efficiency of the corresponding algorithm (see Table 4.4). We recommend these control parameter combinations as starting points for practitioners applying the map L-systems-based method to topology optimization problems.

Later, we picked four representative control parameter combinations from the Pareto front and studied whether we can see similar trends of effectiveness and efficiency on two ‘real-life’ engineering applications. These applications were two different problem setups of the conductive heat transfer problem³ (Section 6.3.5.1) and the topology optimization problem of the integrally stiffened panel (Section 7.5.2). In all optimization problems, the relative efficiency of the control parameters was similar to the statistical experiment. The relative effectiveness was also similar to the statistical experiment on one of the optimization problems, but noisy or dissimilar on the other two.

Contribution 3: The statistical experiment (Contribution 2) was motivated by our third hypothesis that the L-systems encoding acts as a ‘shield’, mitigating the problem-dependency of the best performing control parameter combinations. Based on the results of the statistical experiment, we also showed that the rankings of the combinations have a strong correlation between the test cases (Spearman’s rank correlation coefficient ranges from 0.645 to 0.979), which indicates that the results have only a weak problem-dependency and, thus, supports the above described hypothesis. Finally, we acknowledge that it remains unknown whether the observed correlation is stronger than what would be obtained from a similar experiment with direct encoding.

Contribution 4: In the literature, one of the design variables of the map L-systems-based method is typically the ordinal of the developmental stage. A minor change in this variable may cause a major change in the phenotype, which is potentially disadvantageous to the evolvability of the method. We proposed to linearize the age variable, and benchmarked the linearized age against the baseline age on the test cases (Chapter 5). The linearization of the age variable both expands the design space and eliminates the evolvability disadvantage of the method. We obtained statistical significance that the algorithm with linearized age is more effective in three out of five and more efficient in four out of five test cases than the corresponding algorithm with the baseline age. We

²These values are averages of Test cases 1-4, as no global optimum was found for Test case 5.

³It should be noted that, here, the interpretation formalism of L-systems was different (the turtle interpretation) to that in the statistical experiment (map L-systems), which may limit the strength of this result.

also ran experiments of the linear and baseline age on the application of the sUAV wing internal structure, but the results were not statistically significant.

8.2 Suitability of generative encodings

The second research aim of this work was to identify engineering design problems to which generative encodings, or more precisely L-systems-based methods, are particularly suitable or unsuitable.

Contribution 5: L-systems and their turtle interpretation are designed to produce bifurcating tree-structures, which have full structural continuity. We tested them as a parameterization method in evolutionary topology optimization of a heat conductor. We obtained statistical significance that the method is more effective than the direct encoding method by [Boichot and Fan \(2016\)](#) in 10 out of 12 tested optimization problems. Moreover, we observed that the method is significantly more efficient – requiring two orders of magnitude fewer objective function evaluations – than the direct encoding method. Further, our results indicate that the method is more effective than the SIMP method by [Marck et al. \(2012\)](#) in optimization problems, the dimensionless coefficient $\phi k_p/k_0$ of which is less or equal to 1 and the objective of which is to minimize the average temperature of the design domain. These results support the hypothesis that the method is suitable for topology optimization of heat conductors.

Contribution 6: We tested the map L-systems-based method in the optimization problem of maximizing the fundamental natural frequency of an integrally stiffened panel (Chapter 7). The method outperformed the traditional iso- and orthogrids designs on the tested optimization problem setup. However, it was slightly less effective than the ground structure approach. Nevertheless, we observed the effectiveness of the ground structure approach to be highly dependent on the a priori defined set of candidate structural members. The map L-systems-based method does not require such a priori definition. In addition, the best known design was obtained by the map L-systems-based method. Both methods facilitate convenient implementation of manufacturing constraints.

Let us now compare the results from the three engineering design problems, which we studied using both L-systems-based methods and the ground structure approach. In the first, i.e. the design of the sUAV wing internal structure (Sections 3.7 and 5.3), the ground structure approach yielded, on average, 56.3% lighter designs than the map L-systems-based method. In the second, i.e. the conductive heat transfer problem, the method based on L-systems and its turtle interpretation yielded designs the average temperatures of which were, on average, 24.1% lower than those obtained by the ground structure approach. In the third, i.e. the stiffener layout problem of the integrally stiffened aluminum panel, the ground structure approach yielded designs

the fundamental natural frequencies of which were, on average, 3.2% higher than those obtained by the map L-systems-based method. In the second and third application, we observed that the results of the ground structure approach were highly dependent on the type and density of the ground structure.

A fundamental difference between the phenotypes generated by the ground structure approach and L-systems-based methods is in the continuity of their material distribution. The ground structure approach flexibly generates discontinuous material distributions, which is favorable in the first application. On the other hand, the L-systems-based methods naturally generate continuous material distributions, which is favorable in the second and third application. We believe that this is the primary reason for the differences between the obtained results.

We studied two interpretation formalisms of L-systems, which were map L-systems (the first and third application) and the turtle interpretation (the second application). One of the benefits of the turtle interpretation is that it enables using more design variables which continuously change the geometry of the phenotype (see parametric symbols in Section 2.7.1.1) than the map L-systems. In the map L-systems-based method, we used in this work⁴, the variation of the design variables causes discrete changes in the geometry of the phenotype. Continuous design variables increase the flexibility of the parameterization.

8.3 On patterns of similar structural members

Finally, let us revert back to the introduction (Chapter 1), where we presented three engineering designs, containing repeating patterns of similar structural members. These repeating patterns are motivated by their optimality, aesthetics and manufacturability.

In the literature, a prime example of these repeating patterns is the aircraft wing design obtained by Aage et al. (2017) (see Figure 2.12). Despite the fact that their design space has enormous flexibility (due to over one billion design variables), their optimization process converged to a design, in which the material distribution near the trailing edge has clear repeating patterns. Presumably, generative encodings could define approximately the same distribution of material using significantly fewer design variables.

We obtained designs with repeating patterns of similar structural members, when using the L-systems-based methods. These patterns are the clearest in designs shown in Figures 4.3(a), 4.3(b), 6.19(h) and 7.10(a).

⁴Additional continuous design variables could be included in the map L-systems-based method by using the dynamic method (see Section 2.7.2).

8.4 Publications

This research has resulted in one accepted journal article, one submitted journal article and two conference papers. These publications are listed in Appendix G.

8.5 Future work

In this section, we present three suggestions of the future work to further improve the performance of the L-systems-based topology optimization and expand its applicability.

8.5.1 Exploring the search algorithms

In this work, we used genetic algorithms (GAs) to evolve L-systems parameterizations, because of their popularity in the dedicated scientific literature. However, there is no proof that GAs are more suitable for the purpose than other evolutionary algorithms. We suggest the exploration of other evolutionary algorithms as future work. Interesting algorithms are for example teaching-learning based optimization (TLBO) (Rao et al., 2011) and genetic programming. TLBO is free of control parameters and shown to outperform a selection of evolutionary algorithms on non-linear benchmark problems (Rao et al., 2012). Genetic programming would enable L-systems to be encoded explicitly, without such numerical representations that are required with GAs.

8.5.2 More complex heat transfer problems

The method based on L-systems (and its turtle interpretation) is promising for heat transfer problems. We suggest the future work to involve the implementation of more realistic design domains and physics. As we indicated in Section 6.4, the method is extensible to three-dimensional design domains by introducing two new symbols, which command the turtle to pitch or roll in the three-dimensional space.

An interesting heat transfer problem, to which the method could be applied, is the design of sensible or latent heat storage tanks. The design of such tanks is multi-objective; the main objectives are to maximize the energy density of the tank and to minimize its charge/discharge time. The L-systems-based method is well-suited for multi-objective problems (see Section 6.3.7).

8.5.3 Varying the optimization problem setups of integrally stiffened panels

In Chapter 7, we obtained a regular stiffener layout⁵ (Figure 7.10(a)) that has higher fundamental natural frequency than the traditional iso- and orthogrids on the tested optimization problem. We suggest the future work to investigate whether such layout is also obtained on other optimization problem setups (when using the map L-systems-based method), and whether these layouts are better than the corresponding iso- and orthogrids. The optimization problem setups should consider variable panel dimensions and mass constraints. The future work should also investigate whether the layout has compromised yield or buckling strength in comparison to the iso- and orthogrids. Potentially, the layout could be used more extensively in integrally stiffened panels that are attached to vibration critical engineering applications.

⁵The design consists of circumferential stages and radial lines of stiffeners.

Appendix A

Details of the statistical experiment in Section 5.1.2

This appendix provides details of the statistical experiment in Section 5.1.2, where we tested the three algorithms with baseline and linear age variables, of which the latter was tested with and without the local search. Each of these algorithms was tested on the five test cases, defined in Chapter 4.

The experiment contains a family of 30 statistical tests, in which the three pairwise combinations of the algorithms were tested in terms of two quantities: the completion rate on finding the global optimum, p_c , and the average objective function evaluations, $\bar{Q}$, on the five test cases. However, for Test Case 5, use the average optimized objective function value $\bar{f}_{\min}$, instead of the completion rate p_c , as its global optimum was not found.

The null hypothesis H_0 in each test is that the mean values of a quantity are similar ($\mu_1 = \mu_2$) with the two compared algorithms, while the alternative hypothesis H_1 is that the mean values are dissimilar ($\mu_1 \neq \mu_2$). We use a significance level of $\alpha_s = 0.05$.

When conducting a family of statistical tests, the probability of Type I error is inflated. We threat the inflated probability of Type I error by examining the step-up false discovery rate (FDR) (Benjamini and Hochberg, 1995). We explained both the inflated probability of Type I error and the use of FDR earlier in Section 3.7.4.4.

Considering completion rates $p_{c,1}$ and $p_{c,2}$ of two algorithms, and their corresponding sample sizes n_1 and n_2 , we determine the p -value for the alternative hypothesis H_1 from the Z-value defined as

$$Z = \frac{\mu - \mu_0}{SE}, \quad (\text{A.1})$$

where μ is the point estimate of the difference in the completion rates ($p_{c,1} - p_{c,2}$) and μ_0 is the null value (0). Further, SE is the standard error of the estimate, defined as

$$SE = \sqrt{\frac{\hat{p}(1 - \hat{p})}{n_1} + \frac{\hat{p}(1 - \hat{p})}{n_2}}, \quad (\text{A.2})$$

where

$$\hat{p} = \frac{p_{c,1}n_1 + p_{c,2}n_2}{n_1 + n_2} \quad (\text{A.3})$$

is the pooled completion rate of the two algorithms (Diez et al., 2012).

Probability distributions of average objective function evaluations, $\bar{Q}$, and optimized objective function values, $\bar{f}_{\min}$ (Test Case 5), are skewed. Thus, in the corresponding statistical tests, we determine the p -values using the nonparametric Mann-Whitney U-test.

The results of the statistical tests are ranked in Table A.1, in decreasing order of significance. In each test, the algorithm having, on average, a better performance is listed as algorithm 1, and the algorithm with a worse performance as algorithm 2. The last two columns show the p -value of the test and the corresponding corrected significance level (Equation 3.9), respectively.

Proceeding in the order $j = 30, 29, \dots, 1$, the first test satisfying Equation 3.9 is the test ranked 24th. Therefore, in tests $j = 1, \dots, 24$, we reject the null hypothesis H_0 and accept the alternative hypothesis H_1 , meaning that the means of the quantities are dissimilar. In tests $j = 25, \dots, 30$, we fail to reject the null hypothesis.

rank (j)	test case	quantity	algorithm 1	algorithm 2	p	$j\alpha/n$
1	3	$\bar{Q}$	linear age	linear age (LS)	3.25E-162	1.667E-03
2	2	$\bar{Q}$	linear age	linear age (LS)	6.48E-119	3.333E-03
3	4	$\bar{Q}$	linear age	linear age (LS)	5.06E-116	5.000E-03
4	1	$\bar{Q}$	linear age	linear age (LS)	9.59E-113	6.667E-03
5	2	$\bar{Q}$	baseline age	linear age (LS)	9.31E-105	8.333E-03
6	3	$\bar{Q}$	baseline age	linear age (LS)	2.11E-104	1.000E-02
7	4	$\bar{Q}$	baseline age	linear age (LS)	3.10E-88	1.167E-02
8	3	$\bar{Q}$	linear age	baseline age	3.70E-82	1.333E-02
9	1	$\bar{Q}$	baseline age	linear age (LS)	2.47E-60	1.500E-02
10	5	$\bar{Q}$	baseline age	linear age (LS)	3.13E-57	1.667E-02
11	5	$\bar{Q}$	linear age	linear age (LS)	7.33E-54	1.833E-02
12	4	p_c	linear age (LS)	baseline age	5.361E-47	2.000E-02
13	4	p_c	linear age	baseline age	8.874E-44	2.167E-02
14	1	$\bar{Q}$	linear age	baseline age	4.33E-32	2.333E-02
15	3	p_c	linear age (LS)	baseline age	1.131E-19	2.500E-02
16	5	$\bar{f}_{\min}$	linear age (LS)	baseline age	2.301E-19	2.667E-02
17	3	p_c	linear age (LS)	linear age	2.560E-15	2.833E-02
18	2	p_c	linear age (LS)	baseline age	8.033E-12	3.000E-02
19	5	$\bar{f}_{\min}$	linear age	baseline age	1.434E-09	3.167E-02
20	2	p_c	linear age (LS)	linear age	4.491E-05	3.333E-02
21	4	$\bar{Q}$	linear age	baseline age	1.52E-04	3.500E-02
22	5	$\bar{f}_{\min}$	linear age (LS)	linear age	5.956E-04	3.667E-02
23	2	$\bar{Q}$	linear age	baseline age	1.24E-03	3.833E-02
24	2	p_c	linear age	baseline age	4.548E-03	4.000E-02
25	1	p_c	linear age	baseline age	5.684E-02	4.167E-02
26	1	p_c	linear age (LS)	baseline age	1.352E-01	4.333E-02
27	3	p_c	linear age	baseline age	2.182E-01	4.500E-02
28	5	$\bar{Q}$	baseline age	linear age	4.35E-01	4.667E-02
29	4	p_c	linear age (LS)	linear age	5.296E-01	4.833E-02
30	1	p_c	linear age	linear age (LS)	6.800E-01	5.000E-02

Table A.1: The family of statistical tests ($j = 1, 2, \dots, 30$) ranked in decreasing order of significance, based on their p -values. In each test, algorithm 1 has, on average of the obtained results, a better performance than algorithm 2.

Appendix B

Optimal map L-systems for Test Cases 2-5

This appendix lists the map L-systems which phenotypes were shown in Figure 4.3. These phenotypes are examples of global optima of Test Cases 2-5.

Test Case 2:

$$\begin{aligned}
 \text{Axiom:} & \quad \omega_0 = DBBB \\
 \text{Rules:} & \quad P_0 : A \rightarrow [-\overleftarrow{B}]D \\
 & \quad P_1 : B \rightarrow D[+\overrightarrow{B}][-B][-\overrightarrow{A}]A \\
 & \quad P_2 : C \rightarrow [+\overrightarrow{C}][-\overrightarrow{A}]CB[-\overleftarrow{A}]C \\
 & \quad P_3 : D \rightarrow D[+\overrightarrow{A}][+A]A \\
 \text{Additional variables:} & \quad n = 4 \\
 & \quad f_a = 0.253516222577
 \end{aligned} \tag{B.1}$$

Test Case 3:

$$\begin{aligned}
 \text{Axiom:} & \quad \omega_0 = BBBB \\
 \text{Rules:} & \quad P_0 : A \rightarrow C[+\overrightarrow{A}][-\overleftarrow{A}]B \\
 & \quad P_1 : B \rightarrow C[+C][+\overleftarrow{A}][+B][-A][-B] \\
 & \quad P_2 : C \rightarrow [-B][+A]B[-C][-A] \\
 & \quad P_3 : D \rightarrow D[-\overrightarrow{B}][-\overleftarrow{D}][-\overrightarrow{C}][-C] \\
 \text{Additional variables:} & \quad n = 6 \\
 & \quad f_a = 0.199228645547
 \end{aligned} \tag{B.2}$$

Test Case 4:

$$\begin{aligned}
 \text{Axiom:} & \quad \omega_0 = DADA \\
 \text{Rules:} & \quad \begin{aligned}
 P_0 : A & \rightarrow [+D][-\overleftarrow{C}]DDA \\
 P_1 : B & \rightarrow ABC[-\overleftarrow{C}][-\overrightarrow{B}][+\overleftarrow{A}] \\
 P_2 : C & \rightarrow [+C][+\overrightarrow{B}][-\overrightarrow{C}]CC \\
 P_3 : D & \rightarrow [-C]D[-B][+\overrightarrow{A}][+\overleftarrow{D}][-\overleftarrow{C}]
 \end{aligned} \\
 \text{Additional variables:} & \quad n = 4 \\
 & \quad f_a = 0.456650213568
 \end{aligned} \tag{B.3}$$

Test Case 5:

$$\begin{aligned}
 \text{Axiom:} & \quad \omega_0 = BDDA \\
 \text{Rules:} & \quad \begin{aligned}
 P_0 : A & \rightarrow D[+\overrightarrow{B}]BBBC \\
 P_1 : B & \rightarrow AB[-\overrightarrow{C}][+\overleftarrow{C}]C \\
 P_2 : C & \rightarrow [-\overleftarrow{C}][+\overrightarrow{A}][+\overrightarrow{A}]AA \\
 P_3 : D & \rightarrow D[-A]DAD
 \end{aligned} \\
 \text{Additional variables:} & \quad n = 5 \\
 & \quad f_a = 0.0617572222021
 \end{aligned} \tag{B.4}$$

Appendix C

Collaboration with Dr. Gilles Marck

Table C.1 provides the distinction of Dr. Gilles Marck and the author's contributions to the work reported in Chapter 6 and in the paper by [Ikonen et al. \(2018\)](#).

contribution	the author	Dr. Gilles Marck
1. Literature review	X	-
2. Ground structure parameterization	X	-
3. L-systems-based parameterization	X	-
4. Development of the element width scaling	-	X
5. Development of the projection	-	X
6. Implementation of the finite volume method	-	X
7. Implementation of single and multi-objective genetic algorithms	X	-
8. Development of the plotting function of material and temperature distributions ⁱ	-	X
9. Development of other postprocessing tools	X	-
10. Generation of results using the ground structure approach	X	-
11. Generation of results using the L-systems-based method	X	-
12. Generation of benchmark results using the SIMP method	-	X
13. Statistical analyses	X	-
14. Preparation of the manuscript: Sections 1 to 3.1 and 4 to 6	X	-
15. Preparation of the manuscript: Sections 3.2 to 3.3	-	X

ⁱ See for example Figures [6.19](#) and [6.21](#)

Table C.1: Contributions to the work reported in Chapter 6.

Appendix D

Detailed description of the projection method used in Section 6.2.3

Once the appropriate correction factor c_{sca} has been determined, the last operation of mapping is to project the scaled L-system elements included within the domain Ω to the design grid, which is a Cartesian grid made of $N_x \times N_y$ square cells. In other words, we identify the design cells with centers lying inside the scaled L-system structure Ω_p . This could be done by invoking a *Shapely* routine (which checks if a point is inside a polygon). However, we observed this approach to be inefficient from a computational point of view, mainly due to the complexity of the domain Ω_p . Consequently, we have implemented another approach where we project each trapezoidal element, intersecting with Ω , separately using following steps:

1. we identify the design cells belonging to the bounding box of the trapezoidal element i (see Figure D.1),
2. for each center point of these cells, generically denoted as P ,
 - (a) we compute the non-dimensional abscissa s of its projection along the trapezoidal middle line

$$s = \frac{\overrightarrow{P_{i,0}P} \cdot \vec{u}}{\|\overrightarrow{P_{i,0}P}\|}, \quad (\text{D.1})$$

where $\vec{u}$ is the unit vector between the points $P_{i,0}$ and $P_{i,1}$. If $0 \leq s \leq 1$, the projection of P is between the points $P_{i,0}$ and $P_{i,1}$ and,

- (b) we compute the width $d_i(s)$ of the trapezoidal element i at the abscissa s using equation

$$d_i(s) = \frac{c_{\text{sca}}}{2} (w_{i,0} + s(w_{i,1} - w_{i,0})). \quad (\text{D.2})$$

- (c) Finally, we assign the design cell to the domain Ω_p , if $0 \leq s \leq 1$ and $\|\overrightarrow{P_{i,0}P} \times \vec{u}\| \leq d(s)$. Otherwise, we assign it to the domain Ω_0 .

Moreover, we omit the testing of design cells that have already been assigned to the domain Ω_p when projecting an earlier trapezoidal element. The tips of an element, shown in Figure D.1, are projected using the same method (each tip also has a trapezoidal shape). Finally, if the aspect ratio of a trapezoidal element i is large, it is divided into several sections in order to reduce the size of the bounding boxes to test (and consequently the number of empty cells).

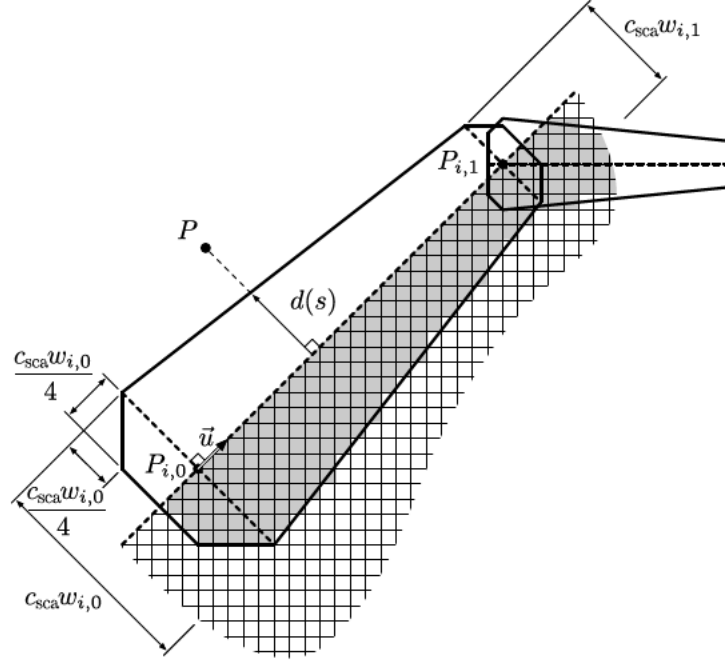
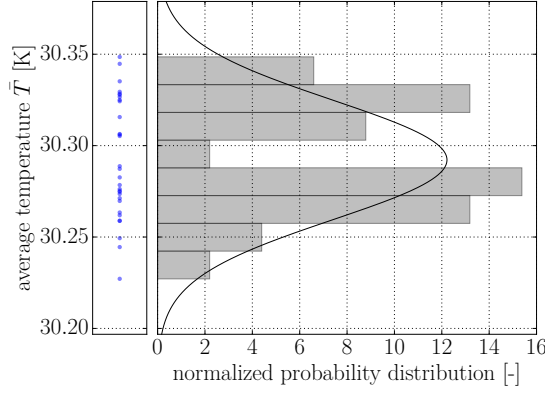


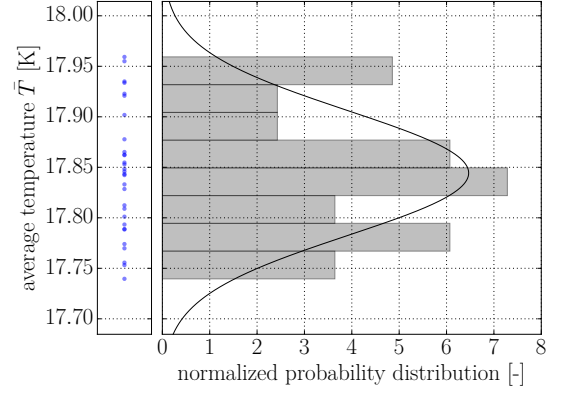
Figure D.1: Projection of an element i , spanning between nodes $P_{i,0}$ and $P_{i,1}$, onto a Cartesian grid.

Appendix E

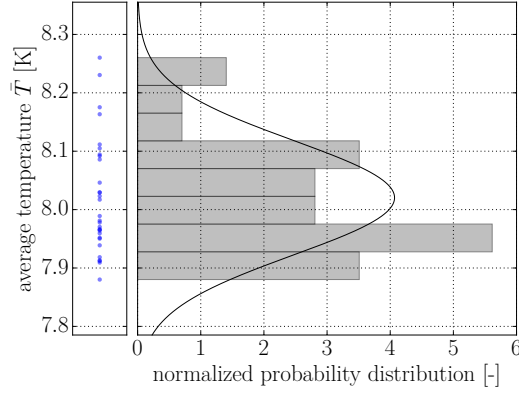
Distributions of optimized objective function values in Section [6.3.6](#)



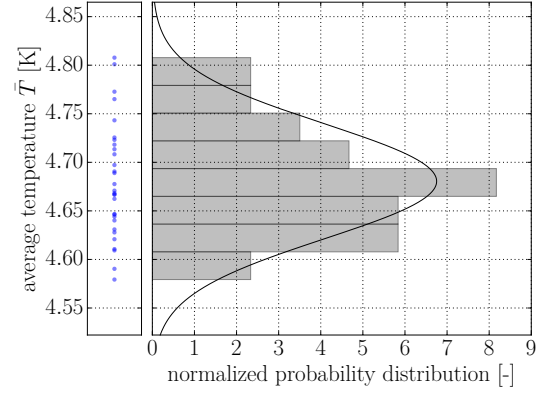
(a) Optimization problem #1



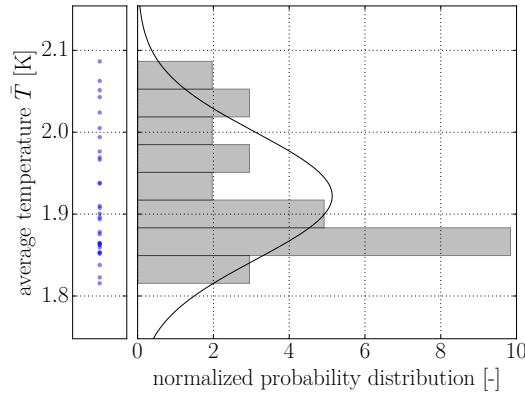
(b) Optimization problem #2



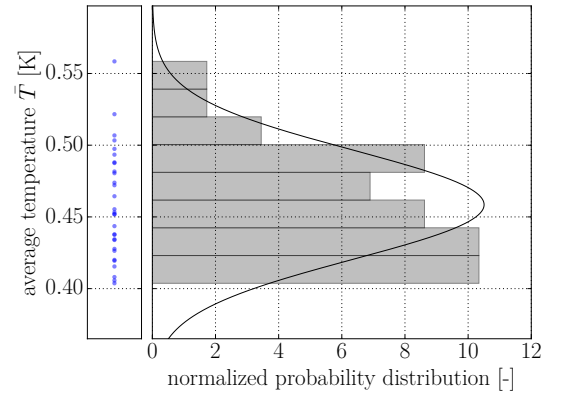
(c) Optimization problem #3



(d) Optimization problem #4

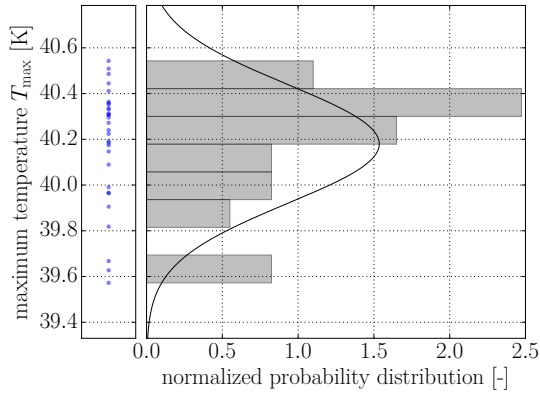


(e) Optimization problem #5

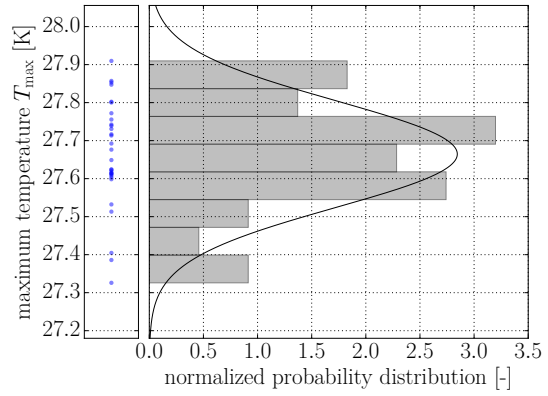


(f) Optimization problem #6

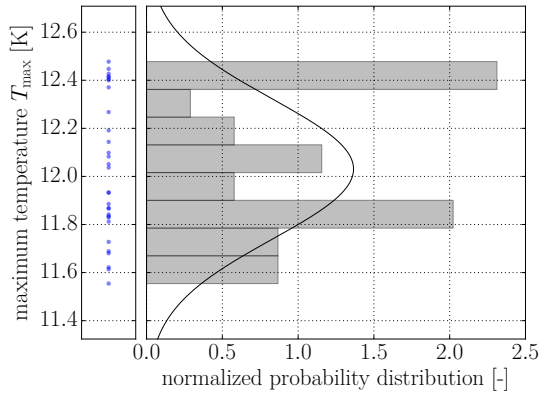
 Figure E.1: Distributions of the optimized average temperature $\bar{T}$ in optimization problems #1-6



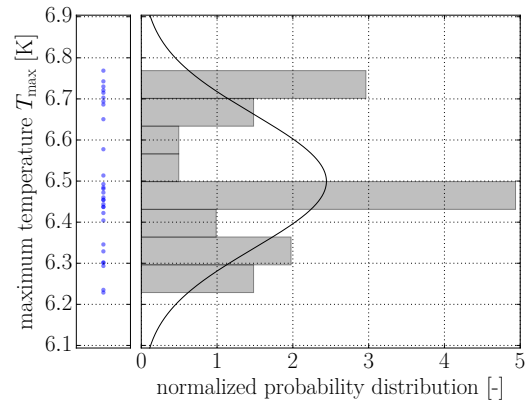
(a) Optimization problem #7



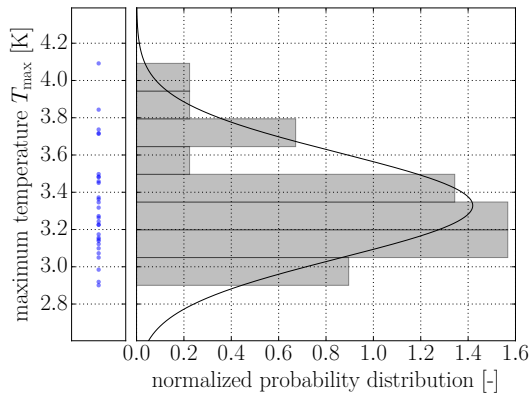
(b) Optimization problem #8



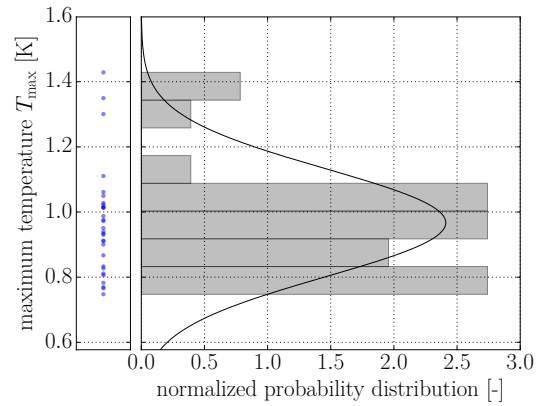
(c) Optimization problem #9



(d) Optimization problem #10



(e) Optimization problem #11

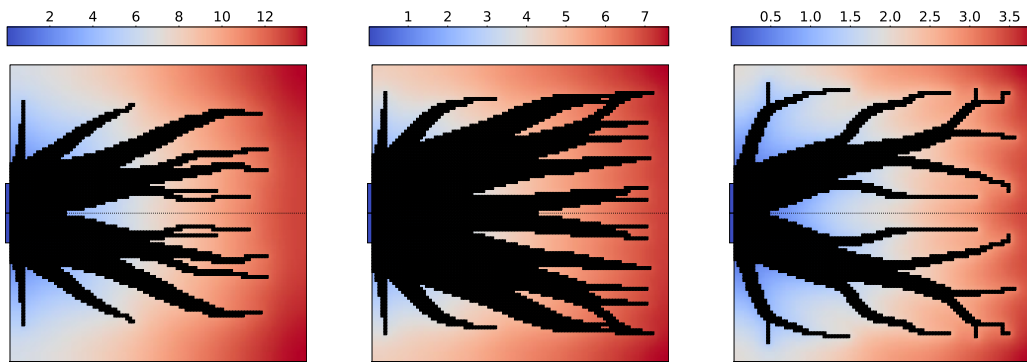


(f) Optimization problem #12

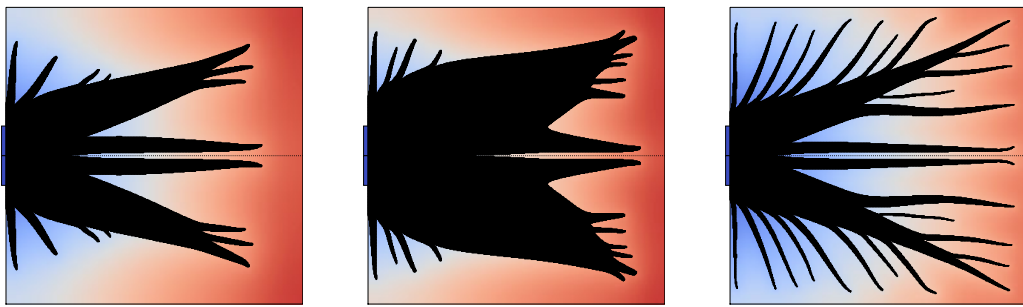
Figure E.2: Distributions of the optimized maximum temperature $T_{\max}$ in optimization problems #7-12

Appendix F

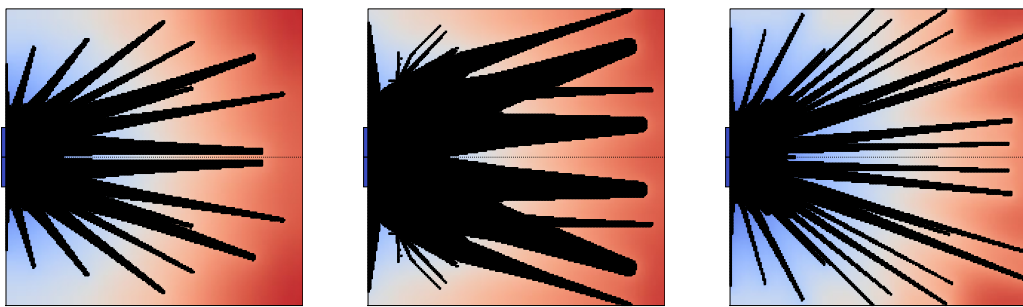
Optimized designs for
optimization problems 3-5 and
9-11 in Section [6.3.6](#)



(a) Problem #3 (direct encoding) (b) Problem #4 (direct encoding) (c) Problem #5 (direct encoding)



(d) Problem #3 (SIMP) (e) Problem #4 (SIMP) (f) Problem #5 (SIMP)



(g) Problem #3 (L-systems) (h) Problem #4 (L-systems) (i) Problem #5 (L-systems)

Figure F.1: Comparison of obtained designs for optimization problems #3, #4 and #5. The unit of the contour map is K.

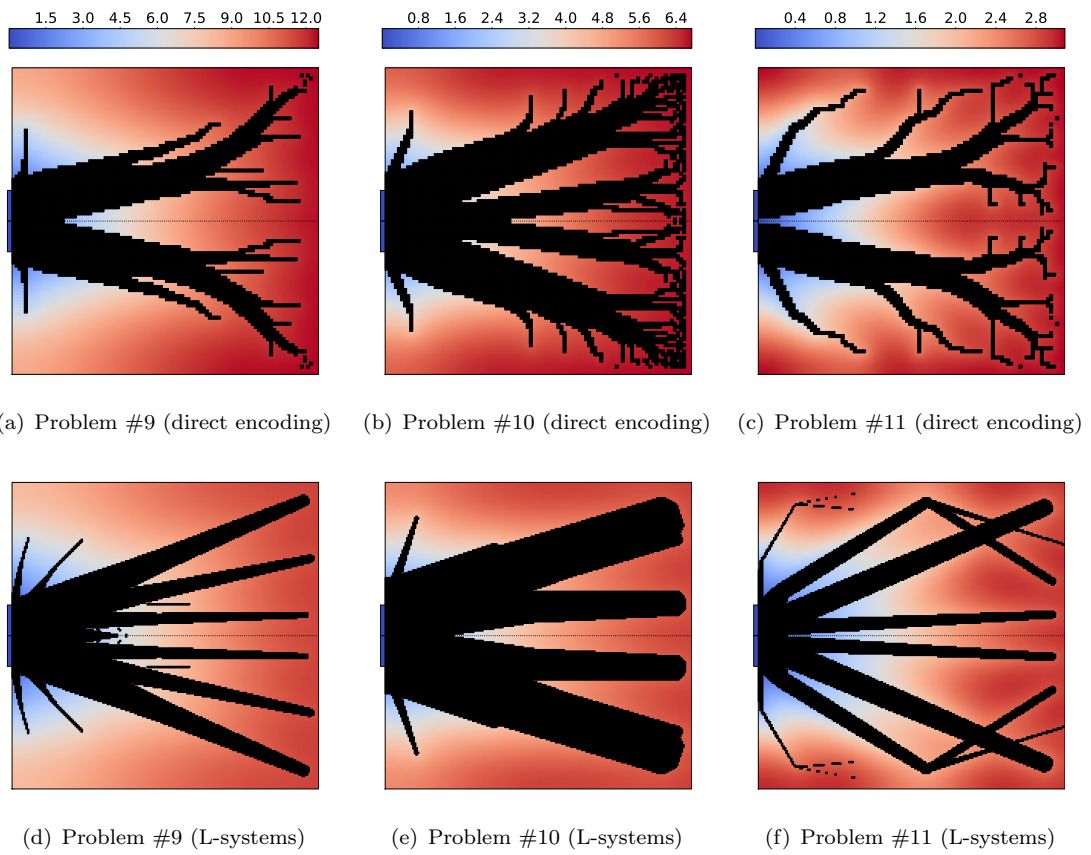


Figure F.2: Comparison of obtained designs for optimization problems #9, #10 and #11. The unit of the contour map is K.

Appendix G

List of publications

This research project has resulted in the following publications:

1. T. J. Ikonen, G. Marck, A. Sóbester, and A. J. Keane. Topology optimization of conductive heat transfer problems using parametric L-systems. *Structural and Multidisciplinary Optimization*, <https://doi.org/10.1007/s00158-018-2055-7>, 2018.
2. T. J. Ikonen and A. Sóbester. Statistical analysis of control parameters in evolutionary map L-systems-based topology optimization. *Structural and Multidisciplinary Optimization*, 58(3):997-1013, 2018.
3. T. J. Ikonen and A. Sóbester. Two variations to the map L-systems-based topology optimization method. In *Proceedings of the 17th AIAA Aviation Technology, Integration, and Operations Conference*, Denver, United States, 2017.
4. T. J. Ikonen and A. Sóbester. Ground structure approaches for the evolutionary optimization of aircraft wing structures. In *Proceedings of the 16th AIAA Aviation Technology, Integration, and Operations Conference*, Washington DC, United States, 2016.

Bibliography

- N. Aage, E. Andreassen, B. S. Lazarov, and O. Sigmund. Giga-voxel computational morphogenesis for structural design. *Nature*, 550(7674):84, 2017.
- A. Agresti and B. A. Coull. Approximate is better than exact for interval estimation of binomial proportions. *The American Statistician*, 52(2):119–126, 1998.
- W. Akl, A. El-Sabbagh, and A. Baz. Optimization of the static and dynamic characteristics of plates with isogrid stiffeners. *Finite Elements in Analysis and Design*, 44(8): 513–523, 2008.
- G. Allaire and F. Jouve. A level-set method for vibration and multiple loads structural optimization. *Computer methods in applied mechanics and engineering*, 194(30):3269–3290, 2005.
- J. T. Allison, A. Khetan, and D. J. Lohan. Managing variable-dimension structural optimization problems using generative algorithms. In *Proceedings of the 10th World Congress on Structural and Multidisciplinary Optimization (WCSMO)*, Orlando, FL, 2013.
- C. A. Anderson, K. F. Jones, and J. Ryan. A two-dimensional genetic algorithm for the ising problem. *Complex Systems*, 5(3):327–334, 1991.
- A. Arcuri and L. Briand. A hitchhiker’s guide to statistical tests for assessing randomized algorithms in software engineering. *Software Testing, Verification and Reliability*, 24(3):219–250, 2014.
- A. Asadpoure, J. K. Guest, and L. Valdevit. Incorporating fabrication cost into topology optimization of discrete structures and lattices. *Structural and Multidisciplinary Optimization*, 51(2):385–396, 2015.
- N. Aulig and M. Olhofer. Evolutionary computation for topology optimization of mechanical structures: An overview of representations. In *Evolutionary Computation (CEC), 2016 IEEE Congress on*, pages 1948–1955. IEEE, 2016.
- J. E. Baker. Reducing bias and inefficiency in the selection algorithm. In *Proceedings of the second international conference on genetic algorithms*, pages 14–21, 1987.

- V. Balabanov and R. T. Haftka. Topology optimization of transport wing internal structure. *Journal of Aircraft*, 33(1):232–233, 1996.
- B. S. Beck, K. A. Cunefare, M. Collet, and M. Ruzzene. Active vibration control of a stiffened panel through application of negative capacitance shunts. In *Active and Passive Smart Structures and Integrated Systems 2011*, volume 7977, page 79770C. International Society for Optics and Photonics, 2011.
- A. Bejan. Constructal-theory network of conducting paths for cooling a heat generating volume. *International Journal of Heat and Mass Transfer*, 40(4):799813–811816, 1997.
- M. P. Bendsøe. Optimal shape design as a material distribution problem. *Structural optimization*, 1(4):193–202, 1989.
- M. P. Bendsøe and N. Kikuchi. Generating optimal topologies in structural design using a homogenization method. *Computer methods in applied mechanics and engineering*, 71(2):197–224, 1988.
- M. P. Bendsøe and O. Sigmund. *Topology optimization: theory, methods and applications*. Springer, Berlin, 2003.
- Y. Benjamini and Y. Hochberg. Controlling the false discovery rate: a practical and powerful approach to multiple testing. *Journal of the royal statistical society. Series B (Methodological)*, pages 289–300, 1995.
- T. Blickle and L. Thiele. A comparison of selection schemes used in genetic algorithms, 1995.
- R. Boichot and Y. Fan. A genetic algorithm for topology optimization of area-to-point heat conduction problem. *International Journal of Thermal Sciences*, 108:209–217, 2016.
- R. Boichot, L. Luo, and Y. Fan. Tree-network structure generation for heat conduction by cellular automaton. *Energy Conversion and Management*, 50(2):376–386, 2009.
- C. Brampton, H. A. Kim, and J. Cuningham. Level set topology optimisation of aircraft wing considering aerostructural interaction. In *12th AIAA Aviation Technology, Integration, and Operations (ATIO) Conference and 14th AIAA/ISSM Multidisciplinary Analysis and Optimization Conference, Indianapolis, IN, September*, volume 17, page 19, 2012.
- P. A. Browne, C. Budd, N. I. M. Gould, H. A. Kim, and J. A. Scott. A fast method for binary programming using first-order derivatives, with application to topology optimization with buckling constraints. *International Journal for Numerical Methods in Engineering*, 92(12):1026–1043, 2012.

- T. N. Bui and B. R. Moon. On multi-dimensional encoding/crossover. In *Proceedings of the 6th International Conference on Genetic Algorithms*, pages 49–56. Morgan Kaufmann Publishers Inc., 1995.
- F. H. Burger, J. Dirker, and J. P. Meyer. Three-dimensional conductive heat transfer topology optimisation in a cubic domain for the volume-to-surface problem. *International Journal of Heat and Mass Transfer*, 67:214–224, 2013.
- Y. Chen and R. F. Gibson. Analytical and experimental studies of composite isogrid structures with integral passive damping. *Mechanics of Advanced Materials and Structures*, 10(2):127–143, 2003.
- Y.-M. Chen, A. Bhaskar, and A. Keane. A parallel nodal-based evolutionary structural optimization algorithm. *Structural and multidisciplinary optimization*, 23(3):241–251, 2002.
- N. Cheney, E. Ritz, and H. Lipson. Automated vibrational design and natural frequency tuning of multi-material structures. In *Proceedings of the 2014 Annual Conference on Genetic and Evolutionary Computation*, pages 1079–1086. ACM, 2014.
- X. Cheng, Z. Li, and Z. Guo. Constructs of highly effective heat transport paths by bionic optimization. *Science in China Series E: Technological Sciences*, 46(3):296–302, 2003.
- N. Chomsky. Three models for the description of language. *IRE Transactions on Information Theory*, 2(3):113–124, 1956.
- T.-Y. Chou, T.-K. Liu, C.-N. Lee, and C.-R. Jeng. Method of inequality-based multiobjective genetic algorithm for domestic daily aircraft routing. *IEEE Transactions on Systems, Man and Cybernetics, Part A: Systems and Humans*, 38(2):299–308, 2008.
- D. N. Chu, Y. M. Xie, A. Hira, and G. P. Steven. Evolutionary structural optimization for problems with stiffness constraints. *Finite Elements in Analysis and Design*, 21(4):239–251, 1996.
- C. A. Coello Coello. Theoretical and numerical constraint-handling techniques used with evolutionary algorithms: a survey of the state of the art. *Computer methods in applied mechanics and engineering*, 191(11):1245–1287, 2002.
- C. A. Coello Coello, G. B. Lamont, and D. A. Van Veldhuizen. *Evolutionary algorithms for solving multi-objective problems*, volume 5. Springer, 2007.
- J. P. Cohoon and W. D. Paris. Genetic placement. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 6(6):956–964, 1987.
- G. W. Corder and D. I. Foreman. *Nonparametric statistics: A step-by-step approach*. John Wiley & Sons, 2014.

- R. Courant. Variational methods for the solution of problems of equilibrium and vibrations. *Bull. Am. Math. Soc.*, 49:1–23, 1943.
- T. Dbouk. A review about the engineering design of optimal heat transfer systems using topology optimization. *Applied Thermal Engineering*, 112:841–854, 2017.
- K. A. De Jong. Analysis of the behavior of a class of genetic adaptive systems. 1975.
- J. D. Deaton and R. V. Grandhi. A survey of structural and multidisciplinary continuum topology optimization: post 2000. *Structural and Multidisciplinary Optimization*, 49(1):1–38, 2014.
- K. Deb and S. Gulati. Design of truss-structures for minimum weight using genetic algorithms. *Finite elements in analysis and design*, 37(5):447–465, 2001.
- K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan. A fast and elitist multiobjective genetic algorithm: Nsga-ii. *IEEE transactions on evolutionary computation*, 6(2):182–197, 2002.
- E. M Dede. Multiphysics topology optimization of heat transfer and fluid flow systems. In *proceedings of the COMSOL Users Conference*, 2009.
- A. R. Díaz and N. Kikuchi. Solutions to shape and topology eigenvalue optimization problems using a homogenization method. *International Journal for Numerical Methods in Engineering*, 35(7):1487–1502, 1992.
- A. R. Díaz and O. Sigmund. Checkerboard patterns in layout optimization. *Structural optimization*, 10(1):40–45, 1995.
- D. M. Diez, C. D. Barr, and M. Cetinkaya-Rundel. *OpenIntro statistics*. 2012.
- X. Ding and K. Yamazaki. Stiffener layout design for plate structures by growing and branching tree model (application to vibration-proof design). *Structural and Multidisciplinary Optimization*, 26(1):99–110, 2004.
- J. Dirker and J. P. Meyer. Topology optimization for an internal heat-conduction cooling scheme in a square domain for high heat flux applications. *Journal of Heat Transfer*, 135(11):111010, 2013.
- M. Dorigo and L. M. Gambardella. Ant colonies for the travelling salesman problem. *BioSystems*, 43(2):73–81, 1997.
- W. S. Dorn, R. E. Gomory, and H. J. Greenberg. Automatic design of optimal structures. *Journal de mecanique*, 3:25–52, 1964.
- J. Du and N. Olhoff. Topological design of freely vibrating continuum structures for maximum values of simple and multiple eigenfrequencies and frequency gaps. *Structural and Multidisciplinary Optimization*, 34(2):91–110, 2007.

- P. D. Dunning, B. K. Stanford, and H. A. Kim. Aerostructural level set topology optimization for a common research model wing. In *AIAA SciTech Conference*, pages 1–21, 2014.
- R. Eberhart and J. Kennedy. A new optimizer using particle swarm theory. In *Proceedings of the Sixth International Symposium on Micro Machine and Human Science, MHS'95.*, pages 39–43. IEEE, 1995.
- S. M. El-Soudani. Methods of making integrally stiffened axial load carrying skin panels for primary aircraft structure and fuel tank structures, 08 2006.
- H. A. Eschenauer and N. Olhoff. Topology optimization of continuum structures: A review*. *Applied Mechanics Reviews*, 54(4):331–390, 2001.
- J. Eves, V. V. Toropov, H. M. Thompson, P. H. Gaskell, J. J. Doherty, and J. Harris. Topology optimization of aircraft with non-conventional configurations. In *Proceedings of the 8th World Congress on Structural and Multidisciplinary Optimization*, 2009.
- A. V. Fiacco and G. P. McCormick. Asymptotic conditions for constrained minimization. Technical report, DTIC Document, 1968.
- R. Fletcher. *Practical methods of optimization*. John Wiley & Sons, 2013.
- F.-A. Fortin, Fr.-M. De Rainville, M.-A. Gardner, M. Parizeau, and C. Gagné. DEAP: Evolutionary algorithms made easy. *Journal of Machine Learning Research*, 13:2171–2175, jul 2012.
- W. E. Frazier. Metal additive manufacturing: a review. *Journal of Materials Engineering and Performance*, 23(6):1917–1928, 2014.
- H. Gagnon and D. W. Zingg. Geometry generation of complex unconventional aircraft with application to high-fidelity aerodynamic shape optimization. In *21st AIAA Computational Fluid Dynamics Conference, No. AIAA-2013-2850, San Diego, CA, United States*, 2013.
- A. H. Gandomi, S. Talatahari, X.-S. Yang, and S. Deb. Design optimization of truss structures using cuckoo search algorithm. *The Structural Design of Tall and Special Buildings*, 22(17):1330–1349, 2013.
- G. Gao, Z.-Y. Liu, Y.-B. Li, and Y.-F. Qiao. A new method to generate the ground structure in truss topology optimization. *Engineering Optimization*, 49(2):235–251, 2017.
- T. Gao, W. H. Zhang, J. H. Zhu, Y. J. Xu, and D. H. Bassir. Topology optimization of heat conduction problem involving design-dependent heat load effect. *Finite Elements in Analysis and Design*, 44(14):805–813, 2008.

- P. Gardonio and S. J. Elliott. Active control of structure-borne and airborne sound transmission through double panel. *Journal of Aircraft*, 36(6):1023–1032, 1999.
- M. Gen and R. Cheng. *Genetic algorithms and engineering optimization*, volume 7. John Wiley & Sons, 2000.
- A. Gersborg-Hansen, M. P. Bendsøe, and O. Sigmund. Topology optimization of heat conduction problems using the finite volume method. *Structural and multidisciplinary optimization*, 31(4):251–259, 2006.
- L. Ghodoossi and N. Egrican. Exact solution for cooling of electronics using constructal theory. *Journal of Applied Physics*, 93(8):4922–4929, 2003.
- M. Giger and P. Ermanni. Evolutionary truss topology optimization using a graph-based parameterization concept. *Structural and Multidisciplinary Optimization*, 32(4):313–326, 2006.
- M. Gilbert and A. Tyas. Layout optimization of large-scale pin-jointed frames. *Engineering Computations*, 20(8):1044–1064, 2003.
- S. Gillies et al. [Shapely: manipulation and analysis of geometric objects](#), 2007.
- J. R. Gloudemans, P. C. Davis, and P. A. Gelhausen. A rapid geometry modeler for conceptual aircraft. In *34th Aerospace Sciences Meeting and Exhibit, Reno, NV Jan*, pages 15–18, 1996.
- D. E. Golberg. Genetic algorithms in search, optimization, and machine learning. *Addison wesley*, 1989, 1989.
- D. E. Goldberg and M. P. Samtani. Engineering optimization via genetic algorithm. In *Electronic computation*, pages 471–482. ASCE, 1986.
- J. Gray, K. T. Moore, and B. A. Naylor. Openmdao: An open source framework for multidisciplinary analysis and optimization. In *13th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference, Fort Worth, TX, AIAA, AIAA-2010-9101*, pages 5–7, 2010.
- D. E. Grierson and W. H. Pak. Optimal sizing, geometrical and topological design using a genetic algorithm. *Structural Optimization*, 6(3):151–159, 1993.
- F. Gruau. Genetic synthesis of modular neural networks. In *Proceedings of the 5th International Conference on Genetic Algorithms*, pages 318–325. Morgan Kaufmann Publishers Inc., 1993.
- F. Gruau. *Neural Network Synthesis using Cellular Encoding and the Genetic Algorithm*. PhD thesis, Ecole Normale Supérieure de Lyon, 15 parvis Ren Descartes, 69342 Lyon Cedex 07, France, 1994.

- T. Guo, D. J. Lohan, R. Cang, M. Y. Ren, and J. T. Allison. An indirect design representation for topology optimization using variational autoencoder and style transfer. In *2018 AIAA/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference*, page 0804, 2018.
- A. S. Hahn. Vehicle sketch pad: a parametric geometry modeler for conceptual aircraft design. In *48th AIAA Aerospace Sciences Meeting*, pages 2010–657, 2010.
- H. Hamda and M. Schoenauer. Topological optimum design with evolutionary algorithms. *Journal of Convex Analysis*, 9(2):503–518, 2002.
- L. U. Hansen and P. Horst. Multilevel optimization in aircraft structural design evaluation. *Computers & structures*, 86(1):104–118, 2008.
- L. He and M. Gilbert. Rationalization of trusses generated via layout optimization. *Structural and Multidisciplinary Optimization*, 52(4):677–694, 2015.
- J. H. Holland. *Adaptation in natural and artificial systems*. University of Michigan Press, 1975.
- G. S. Hornby and J. B. Pollack. The advantages of generative grammatical encodings for physical design. In *Proceedings of the 2001 Congress on Evolutionary Computation*, volume 1, pages 600–607. IEEE, 2001.
- G. S. Hornby and J. B. Pollack. Creating high-level components with a generative representation for body-brain evolution. *Artificial Life*, 8(3):223–246, 2002.
- X. Huang and M. Xie. *Evolutionary topology optimization of continuum structures: methods and applications*. John Wiley & Sons, 2010a.
- X. Huang and Y. M. Xie. Convergent and mesh-independent solutions for the bi-directional evolutionary structural optimization method. *Finite Elements in Analysis and Design*, 43(14):1039–1049, 2007.
- X. Huang and Y.-M. Xie. A further review of eso type methods for topology optimization. *Structural and Multidisciplinary Optimization*, 41(5):671–683, 2010b.
- X. Huang, Z. H. Zuo, and Y. M. Xie. Evolutionary topological optimization of vibrating continuum structures for natural frequencies. *Computers & structures*, 88(5):357–364, 2010.
- S. M. Huybrechts, S. E. Hahn, and T. E. Meink. Grid stiffened structures: a survey of fabrication, analysis and design methods. In *Proceedings of the 12th international conference on composite materials (ICCM/12)*, 1999.
- S. M. Huybrechts, T. E. Meink, P. M. Wegner, and J. M. Ganley. Manufacturing theory for advanced grid stiffened structures. *Composites Part A: Applied Science and Manufacturing*, 33(2):155–161, 2002.

- J. Hwang and J. R. R. A. Martins. Geomach: Geometry-centric mdao of aircraft configurations with high fidelity. In *Proceedings of the 14th AIAA/ISSMO Multidisciplinary Analysis Optimization Conference, Indianapolis, IN*, 2012.
- T. J. Ikonen, G. Marck, A. Sóbester, and A. J. Keane. Topology optimization of conductive heat transfer problems using parametric L-systems. *Structural and Multidisciplinary Optimization*, <https://doi.org/10.1007/s00158-018-2055-7>, 2018.
- T. J. Ikonen and A. Sóbester. Ground structure approaches for the evolutionary optimization of aircraft wing structures. In *Proceedings of the 16th AIAA Aviation Technology, Integration, and Operations Conference, Washington DC, United States*, 2016.
- T. J. Ikonen and A. Sóbester. Two variations to the map L-systems-based topology optimization method. In *Proceedings of the 17th AIAA Aviation Technology, Integration, and Operations Conference, Denver, United States*, 2017.
- T. J. Ikonen and A. Sóbester. Statistical analysis of control parameters in evolutionary map L-systems-based topology optimization. *Structural and Multidisciplinary Optimization*, 58(3):997–1013, 2018.
- K. Imai and L. A. Schmit. Configuration optimization of trusses. *Journal of the Structural Division*, 107(5):745–756, 1981.
- K. Inoue, M. Yamanaka, and M. Kihara. Optimum stiffener layout for the reduction of vibration and noise of gearbox housing. *Journal of Mechanical Design*, 124(3): 518–523, 2002.
- P. L. Jakab. Wood to metal: The structural origins of the modern airplane. *Journal of Aircraft*, 36(6):914–918, 1999.
- K. A. James and J. R. R. A. Martins. An isoparametric approach to level set topology optimization using a body-fitted finite-element mesh. *Computers & Structures*, 90: 97–106, 2012.
- C. Z. Janikow and Z. Michalewicz. An experimental comparison of binary and floating point representations in genetic algorithms. In *ICGA*, pages 31–36, 1991.
- J. Ji, X. Ding, and M. Xiong. Optimal stiffener layout of plate/shell structures by bionic growth method. *Computers & Structures*, 135:88–99, 2014.
- J. Joines and C. R. Houck. On the use of non-stationary penalty functions to solve nonlinear constrained optimization problems with ga's. In *Evolutionary Computation, 1994. IEEE World Congress on Computational Intelligence., Proceedings of the First IEEE Conference on*, pages 579–584. IEEE, 1994.

- S. Jung and B.-R. Moon. Toward minimal restriction of genetic encoding and crossovers for the two-dimensional euclidean tsp. *Evolutionary Computation, IEEE Transactions on*, 6(6):557–565, 2002.
- C. V. Jutte, B. K. Stanford, C. D. Wieseman, and J. B. Moore. Aeroelastic tailoring of the nasa common research model via novel material and structural configurations. In *AIAA SciTech Conference*, pages 13–17, 2014.
- A. B. Kahng and B.-R. Moon. Toward more powerful recombinations. In *Proceedings of the 6th International Conference on Genetic Algorithms*, pages 96–103, 1995.
- A. Kaveh and A. Zolghadr. Comparison of nine meta-heuristic algorithms for optimal design of truss structures with frequency constraints. *Advances in Engineering Software*, 76:9–30, 2014.
- A. J. Keane and P. Nair. *Computational approaches for aerospace design: the pursuit of excellence*. John Wiley & Sons, 2005.
- L. Kelly, A. J. Keane, A. Sobester, and D. J. J. Toal. Topology optimisation: Increasing the speed and reliability of design. In *15th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference, AIAA Aviation. American Institute of Aeronautics and Astronautics*, 2014.
- A. Khetan, D. J. Lohan, and J. T. Allison. Managing variable-dimension structural optimization problems using generative algorithms. *Structural and Multidisciplinary Optimization*, 52(4):695–715, 2015.
- J. Kim, I. Hwang, Y.-H. Kim, and B.-R. Moon. Genetic approaches for graph partitioning: a survey. In *Proceedings of the 13th annual Conference on Genetic and Evolutionary Computation*, pages 473–480, 2011.
- T. S. Kim and Y. Y. Kim. Mac-based mode-tracking in structural topology optimization. *Computers & Structures*, 74(3):375–383, 2000.
- D. P. Kingma and M. Welling. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013.
- S. Kirkpatrick, C. D. Gelatt, and M. P. Vecchi. Optimization by simulated annealing. *science*, 220(4598):671–680, 1983.
- M. H. Kobayashi. On a biologically inspired topology optimization method. *Communications in Nonlinear Science and Numerical Simulation*, 15(3):787–802, 2010.
- M. H. Kobayashi, A. LeBon, H. T. C. Pedro, R. M. Kolonay, and G. W. Reich. On a cellular division model for multi-disciplinary optimization. In *51st AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference*, page 2989, 2010.

- M. H. Kobayashi, H. T. C. Pedro, R. M. Kolonay, and G. W. Reich. On a cellular division method for aircraft structural design. *Aeronautical Journal*, 113(1150):821–831, 2009.
- R. M. Kolonay and M. H. Kobayashi. 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*, 2010.
- R. M. Kolonay and M. H. Kobayashi. Optimization of aircraft lifting surfaces using a cellular division method. *Journal of Aircraft*, 52(6):2051–2063, 2015.
- A. Konak, D. W. Coit, and A. E. Smith. Multi-objective optimization using genetic algorithms: A tutorial. *Reliability Engineering & System Safety*, 91(9):992–1007, 2006.
- J. R. Koza. *Genetic programming: on the programming of computers by means of natural selection*, volume 1. MIT press, 1992.
- J. R. Koza. *Genetic programming II: automatic discovery of reusable programs*. MIT press, 1994.
- A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105, 2012.
- L. Krog, A. Tucker, M. Kemp, and R. Boyd. Topology optimization of aircraft wing box ribs. In *10th AIAA/ISSMO multidisciplinary analysis and optimization conference*, pages 1–11, 2004.
- B. M. Kulfan and J. E. Bussioletti. Fundamental parametric geometry representations for aircraft component shapes. In *11th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference*, pages 1–42. sn, 2006.
- M. Langelaar. Engineering: Computational design hits record resolution. *Nature*, 550(7674):50, 2017.
- A. Lencus, O. M. Querin, G. P. Steven, and Y. M. Xie. Aircraft wing design automation with eso and geso. *International journal of vehicle design*, 28(4):356–369, 2002.
- B. Li, J. Hong, and Z. Liu. Stiffness design of machine tool structures by a biologically inspired topology optimization method. *International Journal of Machine Tools and Manufacture*, 84:33–44, 2014.
- H. Li, R. V. Grandhi, and M. Kobayashi. Level-set based cellular division method for structural shape and topology optimization. In *2018 AIAA/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference*, page 1387, 2018.

- Q. Li, G. P. Steven, O. M. Querin, and Y. M. Xie. Shape and topology design for heat conduction by evolutionary structural optimization. *International Journal of Heat and Mass Transfer*, 42(17):3361–3371, 1999a.
- Q. Li, G. P. Steven, and Y. M. Xie. On equivalence between stress criterion and stiffness criterion in evolutionary structural optimization. *Structural optimization*, 18(1):67–73, 1999b.
- Q. Q. Liang. *Performance-based Optimization of Structures: Theory and applications*. CRC Press, 2005.
- A. Lindenmayer. Mathematical models for cellular interactions in development i. filaments with one-sided inputs. *Journal of theoretical biology*, 18(3):280–299, 1968a.
- A. Lindenmayer. Mathematical models for cellular interactions in development ii. simple and branching filaments with two-sided inputs. *Journal of theoretical biology*, 18(3):300–315, 1968b.
- A. Lindenmayer and G. Rozenberg. Parallel generation of maps: Developmental systems for cell layers. In *International Workshop on Graph Grammars and Their Application to Computer Science*, pages 301–316. Springer, 1978.
- Y. Liu and G. Glass. Effects of mesh density on finite element analysis. Technical report, SAE Technical Paper, 2013.
- D. Locatelli, S. B. Mulani, and R. K. Kapania. Wing-box weight optimization using curvilinear spars and ribs (sparibs). *Journal of Aircraft*, 48(5):1671–1684, 2011.
- D. J. Lohan, E. M. Dede, and J. T. Allison. Topology optimization for heat conduction using generative design algorithms. *Structural and Multidisciplinary Optimization*, 55(3):1063–1077, 2017.
- G.-C. Luh and C.-Y. Lin. Optimal design of truss structures using ant algorithm. *Structural and Multidisciplinary Optimization*, 36(4):365–379, 2008.
- G.-C. Luh and C.-Y. Lin. Optimal design of truss-structures using particle swarm optimization. *Computers & structures*, 89(23-24):2221–2232, 2011.
- T. Lukaczyk, A. D. Wendorff, E. Botero, T. MacDonald, T. Momose, A. Variyar, J. M. Vegh, M. Colonno, T. D. Economon, J. J. Alonso, et al. Suave: An open-source environment for multi-fidelity conceptual vehicle design. In *16th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference*, page 3087, 2015.
- S. Luke and L. Spector. Evolving graphs and networks with edge encoding: Preliminary report. In *Late breaking papers at the genetic programming 1996 conference*, pages 117–124. Stanford University Stanford, CA, 1996.

- Z. Luo, N. Zhang, W. Gao, and H. Ma. Structural shape and topology optimization using a meshless galerkin level set method. *International Journal for Numerical Methods in Engineering*, 90(3):369–389, 2012.
- C.E. Majewski and N. Hopkinson. Effect of section thickness and build orientation on tensile properties and material characteristics of laser sintered nylon-12 parts. *Rapid Prototyping Journal*, 17(3):176–180, 2011.
- D. Manickarajah, Y. M. Xie, and G. P. Steven. An evolutionary method for optimization of plate buckling resistance. *Finite Elements in Analysis and Design*, 29(3):205–230, 1998.
- D. Manickarajah, Y. M. Xie, and G. P. Steven. Optimisation of columns and frames against buckling. *Computers & Structures*, 75(1):45–54, 2000.
- G. Marck. *Topological optimization of heat and mass transfer (application to heat exchangers)*. PhD thesis, MINES ParisTech, 5 rue Lon Blum, 91120 Palaiseau, France, 2012.
- G. Marck, M. Nemer, J.-L. Harion, S. Russeil, and D. Bougeard. Topology optimization using the simp method for multiobjective conductive problems. *Numerical Heat Transfer, Part B: Fundamentals*, 61(6):439–470, 2012.
- D. A. Masters, N. J. Taylor, T. C. S. Rendall, and C. B. Allen. Multilevel subdivision parameterization scheme for aerodynamic shape optimization. *AIAA Journal*, pages 1–16, 2017a.
- D. A. Masters, N. J. Taylor, T. C. S. Rendall, and C. B. Allen. Three-dimensional subdivision parameterisation for aerodynamic shape optimisation. In *55th AIAA Aerospace Sciences Meeting*, page 0036, 2017b.
- F. Mathieu-Potvin and L. Gosselin. Optimal conduction pathways for cooling a heat-generating body: A comparison exercise. *International journal of heat and mass transfer*, 50(15):2996–3006, 2007.
- K. Maute and M. Allen. Conceptual design of aeroelastic structures by topology optimization. *Structural and Multidisciplinary Optimization*, 27(1):27–42, 2004.
- Z. Mei and P. G. Maropoulos. Review of the application of flexible, measurement-assisted assembly technology in aircraft manufacturing. *Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture*, 228(10):1185–1197, 2014.
- A. G. M. Michell. The limits of economy of material in frame-structures. *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science*, 8(47):589–597, 1904.

- L. F. F. Miguel, R. H. Lopez, and L. F. F. Miguel. Multimodal size, shape, and topology optimisation of truss structures using the firefly algorithm. *Advances in Engineering Software*, 56:23–37, 2013.
- M. Mitchell. *An introduction to genetic algorithms*. MIT press, 1998.
- M. Moshrefi-Torbati, A. J. Keane, S. J. Elliott, M. J. Brennan, and E. Rogers. Passive vibration control of a satellite boom structure by geometric optimization using genetic algorithm. *Journal of Sound and Vibration*, 267(4):879–892, 2003.
- D. J. Munk, G. A. Vio, and G. P. Steven. Topology and shape optimization methods using evolutionary algorithms: a review. *Structural and Multidisciplinary Optimization*, 52(3):613–631, 2015.
- S. Nakagawa. A farewell to bonferroni: the problems of low statistical power and publication bias. *Behavioral Ecology*, 15(6):1044–1045, 2004.
- A. Nakamura, A. Lindenmayer, and K. Aizawa. Some systems for map generation. In *The Book of L*, pages 323–332. Springer, 1986.
- M. M. Neves, H. Rodrigues, and J. M. Guedes. Generalized topology design of structures with a buckling load criterion. *Structural Optimization*, 10(2):71–78, 1995.
- M. M. Neves, O. Sigmund, and M. P. Bendsøe. Topology optimization of periodic microstructures with a penalization of highly localized buckling modes. *International Journal for Numerical Methods in Engineering*, 54(6):809–834, 2002.
- J. Nocedal and S. Wright. *Numerical optimization*. Springer Science & Business Media, 2006.
- E. Oktay, H. U. Akay, and O. T. Sehitoglu. Three-dimensional structural topology optimization of aerial vehicles under aerodynamic loads. *Computers & Fluids*, 92:225–232, 2014.
- I. Ono, M. Yamamura, and S. Kobayashi. A genetic algorithm for job-shop scheduling problems using job-based order crossover. In *Proceedings of IEEE International Conference on Evolutionary Computation*, pages 547–552. IEEE, 1996.
- S. Osher and J. A. Sethian. Fronts propagating with curvature-dependent speed: algorithms based on hamilton-jacobi formulations. *Journal of Computational Physics*, 79(1):12–49, 1988.
- S. J. Osher and F. Santosa. Level set methods for optimization problems involving geometry and constraints: I. frequencies of a two-density inhomogeneous drum. *Journal of Computational Physics*, 171(1):272–288, 2001.
- E. W. Parkes. Joints in optimum frameworks. *International Journal of Solids and Structures*, 11(9):1017–1022, 1975.

- N. L. Pedersen. Maximization of eigenvalues using topology optimization. *Structural and Multidisciplinary Optimization*, 20(1):2–11, 2000.
- H. Pedro, M. H. Kobayashi, C. F. M. Coimbra, and A. K. de Silva. Effectiveness of complex design through an evolutionary approach. *Journal of Thermophysics and Heat Transfer*, 22(1):115, 2008.
- H.-T. C. Pedro and M. H. Kobayashi. On a cellular division method for topology optimization. *International Journal for Numerical Methods in Engineering*, 88(11):1175–1197, 2011.
- T. V. Perneger. What’s wrong with bonferroni adjustments. *Bmj*, 316(7139):1236–1238, 1998.
- C. S. Perone. Pyevolve: a python open-source framework for genetic algorithms. *ACM SIGEVOlution*, 4(1):12–20, 2009.
- W. Prager. A note on discretized michell structures. *Computer Methods in Applied Mechanics and Engineering*, 3(3):349–355, 1974.
- T. J. Pritchard, M. Gilbert, and A. Tyas. Plastic layout optimization of large-scale frameworks subject to multiple load cases, member self-weight and with joint length penalties. In *6th World Congress of Structural and Multidisciplinary Optimization*. Citeseer, 2005.
- P. Prusinkiewicz. Graphical applications of l-systems. In *Proceedings of graphics interface*, volume 86, pages 247–253, 1986.
- P. Prusinkiewicz and J. Hanan. Visualization of botanical structures and processes using parametric l-systems. In *Scientific visualization and graphics simulation*, pages 183–201. John Wiley & Sons, Inc., 1990.
- P. Prusinkiewicz and A. Lindenmayer. *The algorithmic beauty of plants*. Springer Science & Business Media, 2012.
- O. M. Querin, G. P. Steven, and Y. M. Xie. Evolutionary structural optimisation (eso) using a bidirectional algorithm. *Engineering Computations*, 15(8):1031–1048, 1998.
- O. M. Querin, V. Young, G. P. Steven, and Y. M. Xie. Computational efficiency and validation of bi-directional evolutionary structural optimisation. *Computer methods in applied mechanics and engineering*, 189(2):559–573, 2000.
- S. D. Rajan. Sizing, shape, and topology design optimization of trusses using genetic algorithm. *Journal of Structural Engineering*, 121(10):1480–1487, 1995.
- R. V. Rao, V. J. Savsani, and D. P. Vakharia. Teaching–learning-based optimization: a novel method for constrained mechanical design optimization problems. *Computer-Aided Design*, 43(3):303–315, 2011.

- R. V. Rao, V. J. Savsani, and D. P. Vakharia. Teaching–learning-based optimization: an optimization method for continuous non-linear large scale problems. *Information Sciences*, 183(1):1–15, 2012.
- I. Rechenberg. Cybernetic solution path of an experimental problem. 1965.
- I. Rechenberg. *Evolutionsstrategie Optimierung technischer Systeme nach Prinzipien der biologischen Evolution*. Frommann-Holzboog (Stuttgart), 1973.
- J. Rieffel, F. Valero-Cuevas, and H. Lipson. Automated discovery and optimization of large irregular tensegrity structures. *Computers & Structures*, 87(5):368–379, 2009.
- J. H. Rong, Y. M. Xie, and X. Y. Yang. An improved method for evolutionary structural optimisation against buckling. *Computers & Structures*, 79(3):253–263, 2001.
- G. I. N. Rozvany. A critical review of established methods of structural topology optimization. *Structural and multidisciplinary optimization*, 37(3):217–237, 2009.
- A. Runions, M. Fuhrer, B. Lane, P. Federl, A.-G. Rolland-Lagan, and P. Prusinkiewicz. Modeling and visualization of leaf venation patterns. *ACM Transactions on Graphics (TOG)*, 24(3):702–711, 2005.
- E. Sabbatini, G. M. Revel, and M. H. Kobayashi. Vibration reduction using biologically inspired topology optimization method: optimal stiffeners distribution on an acoustically excited plate. *Journal of Vibration and Control*, 21(7):1398–1412, 2015.
- A. Sadrzadeh. A genetic algorithm with the heuristic procedure to solve the multi-line layout problem. *Computers & Industrial Engineering*, 62(4):1055–1064, 2012.
- V. J. Savsani, G. G. Tejani, and V. K. Patel. Truss topology optimization with static and dynamic constraints using modified subpopulation teaching–learning-based optimization. *Engineering Optimization*, 48(11):1990–2006, 2016.
- J. D. Schaffer. Multiple objective optimization with vector evaluated genetic algorithms. In *Proceeding of the First International Conference on Genetic Algorithms and Their Applications, 1985*. LawrenceErlbaumAssociates, Inc., Publishers, 1985.
- M. Schoenauer. Shape representations and evolution schemes. *Evolutionary Programming*, 5, 1996.
- H.-P. Schwefel. *Evolutionsstrategie und numerische Optimierung*. Technische Universität Berlin, 1975.
- H.-P. Schwefel. *Numerische optimierung von computer-modellen mittels der evolutionssstrategie*, volume 1. Birkhäuser, Basel Switzerland, 1977.
- T. Sekimoto and H. Noguchi. Novel development and applications in finite element technologies. homologous topology optimization in large displacement and buckling

- problems. *JSME International Journal Series A Solid Mechanics and Material Engineering*, 44(4):616–622, 2001.
- J. A. Sethian and A. Wiegmann. Structural boundary design via level set and immersed interface methods. *Journal of computational physics*, 163(2):489–528, 2000.
- O. Sigmund. A 99 line topology optimization code written in matlab. *Structural and Multidisciplinary Optimization*, 21(2):120–127, 2001.
- O. Sigmund. On the usefulness of non-gradient approaches in topology optimization. *Structural and Multidisciplinary Optimization*, 43(5):589–596, 2011.
- O. Sigmund and J. Petersson. Numerical instabilities in topology optimization: a survey on procedures dealing with checkerboards, mesh-dependencies and local minima. *Structural optimization*, 16(1):68–75, 1998.
- D. Simon. *Evolutionary optimization algorithms*. John Wiley & Sons, 2013.
- T. W. Simpson and B. S. Dsouza. Assessing variable levels of platform commonality within a product family using a multiobjective genetic algorithm. *Concurrent Engineering*, 12(2):119–129, 2004.
- S. N. Sivanandam and S. N. Deepa. *Introduction to genetic algorithms*. Springer Science & Business Media, 2007.
- M. P. Snyder and T. A. Weisshaar. Simultaneous configuration optimization of multi-state reconfigurable aerostructures. *Journal of Aircraft*, 51(3):727–739, 2014.
- A. Sóbester. Four suggestions for better parametric geometries. In *10th AIAA Multidisciplinary Design Optimization Conference*, pages 1–10. American Institute of Aeronautics and Astronautics, 2014.
- A. Sóbester. Self-designing parameter geometries. In *56th AIAA/ASCE/AHS Structures, Structural Dynamics and Material Conference*, 2015.
- A. Sóbester and A. I. J. Forrester. *Aircraft Aerodynamic Design: Geometry and Optimization*. John Wiley & Sons, 2014.
- B. Stanford, P. Beran, and M. H. Kobayashi. Aeroelastic optimization of flapping wing venation: a cellular division approach. *AIAA journal*, 50(4):938–951, 2012.
- B. Stanford, P. Beran, and M. H. Kobayashi. Simultaneous topology optimization of membrane wings and their compliant flapping mechanisms. *AIAA journal*, 51(6):1431–1441, 2013.
- B. Stanford and P. Ifju. Aeroelastic topology optimization of membrane structures for micro air vehicles. *Structural and Multidisciplinary Optimization*, 38(3):301–316, 2009.

- B. K. Stanford and P. D. Dunning. Optimal topology of aircraft rib and spar structures under aeroelastic loads. *Journal of Aircraft*, 52(4):1298–1311, 2014.
- K. O. Stanley. Compositional pattern producing networks: A novel abstraction of development. *Genetic programming and evolvable machines*, 8(2):131–162, 2007.
- K. O. Stanley and R. Miikkulainen. Evolving neural networks through augmenting topologies. *Evolutionary computation*, 10(2):99–127, 2002.
- K. O. Stanley and R. Miikkulainen. A taxonomy for artificial embryogeny. *Artificial Life*, 9(2):93–130, 2003.
- T. Steiner, J. Trommler, M. Brenn, Y. Jin, and B. Sendhoff. Global shape with morphogen gradients and motile polarized cells. In *Evolutionary Computation, 2009. CEC'09. IEEE Congress on*, pages 2225–2232. IEEE, 2009.
- G. N. Stiny. Pictorial and formal aspects of shape and shape grammars and aesthetic systems. 1975.
- K. Svanberg. The method of moving asymptotes a new method for structural optimization. *International journal for numerical methods in engineering*, 24(2):359–373, 1987.
- K. Tai and T. H. Chee. Design of structures and compliant mechanisms by evolutionary optimization of morphological representations of topology. *Journal of Mechanical Design*, 122(4):560–566, 2000.
- D. Tcherniak. Topology optimization of resonating structures using simp method. *International Journal for Numerical Methods in Engineering*, 54(11):1605–1622, 2002.
- L. H. Tenek and I. Hagiwara. Eigenfrequency maximization of plates by optimization of topology using homogenization and mathematical programming. *JSME international journal. Ser. C, Dynamics, control, robotics, design and manufacturing*, 37(4):667–677, 1994.
- André J Torii, Rafael H Lopez, and Leandro FF Miguel. Design complexity control in truss optimization. *Structural and Multidisciplinary Optimization*, 54(2):289–299, 2016.
- N. P. van Dijk, K. Maute, M. Langelaar, and F. Van Keulen. Level-set methods for structural topology optimization: a review. *Structural and Multidisciplinary Optimization*, 48(3):437–472, 2013.
- J. C. Vassberg, M. A. DeHaan, S. M. Rivers, and R. A. Wahls. Development of a common research model for applied cfd validation studies. *AIAA paper*, 6919:2008, 2008.

- S. Y. Wang and K. Tai. Structural topology design optimization using genetic algorithms with a bit-array representation. *Computer methods in applied mechanics and engineering*, 194(36):3749–3770, 2005.
- W. Wang, S. Guo, and W. Yang. Simultaneous partial topology and size optimization of a wing structure using ant colony and gradient based methods. *Engineering Optimization*, 43(4):433–446, 2011.
- T. Weise. Global optimization algorithms – theory and application. *Self-published, 2nd edition*, Available online at <http://www.it-weise.de/projects/bookNew.pdf>, 2009.
- D. Whitley. A genetic algorithm tutorial. *Statistics and Computing*, 4(2):65–85, 1994.
- Y. M. Xie and G. P. Steven. Shape and layout optimization via an evolutionary procedure. In *Proceedings of the International Conference on Computational Engineering Science*, 1992.
- Y. M. Xie and G. P. Steven. A simple approach to structural frequency optimization. *Computers & structures*, 53(6):1487–1491, 1994.
- X. Xu, X. Liang, and J. Ren. Optimization of heat conduction using combinatorial optimization algorithms. *International journal of heat and mass transfer*, 50(9):1675–1682, 2007.
- S. Yan, F. Wang, and O. Sigmund. On the non-optimality of tree structures for heat conduction. *International Journal of Heat and Mass Transfer*, 122:660 – 680, 2018.
- W. Yang, Z. Yue, L. Li, and P. Wang. Aircraft wing structural design optimization based on automated finite element modelling and ground structure approach. *Engineering Optimization*, (1):94–114, 2016.
- X.-S. Yang and S. Deb. Cuckoo search via lévy flights. In *World Congress on Nature & Biologically Inspired Computing, NaBIC*, pages 210–214. IEEE, 2009.
- X. Y. Yang, Y. M. Xie, G. P. Steven, and O. M. Querin. Bidirectional evolutionary method for stiffness optimization. *AIAA journal*, 37(11):1483–1488, 1999.
- O. Yeniay. Penalty function methods for constrained optimization with genetic algorithms. *Mathematical and Computational Applications*, 10(1):45–56, 2005.
- M. Yuan, J. Qiu, H. Ji, W. Zhou, and R. Ohayon. Active control of sound transmission using a hybrid/blind decentralized control approach. *Journal of Vibration and Control*, 21(13):2661–2684, 2015.
- G. R. Zavala, A. J. Nebro, F. Luna, and C. A. Coello. A survey of multi-objective metaheuristics applied to structural optimization. *Structural and Multidisciplinary Optimization*, 49(4):537–558, 2014.

-
- E. Zitzler, K. Deb, and L. Thiele. Comparison of multiobjective evolutionary algorithms: Empirical results. *Evolutionary computation*, 8(2):173–195, 2000.
- E. Zitzler, M. Laumanns, and L. Thiele. Spea2: Improving the strength pareto evolutionary algorithm. 2001.