Review of Industrial Design Optimization by Genetic Algorithms

F. L. Sáez-Gutiérrez[1](✉), F. J. F. Cañavate[1], and A. Guerrero-González[2]

[1] Graphical Engeniering Department, Technical University of Cartagena, Murcia, Spain
francisco.saez@upct.es
[2] Automation and Systems Engeniering Department, Technical University of Cartagena, Murcia, Spain

Abstract. In engineering, genetic algorithms (GA) have been successfully applied to some cases. The current state of this technique has evolved to allow computer designs from a sketch. Thus, GA generate a solution by optimization. Here the final solution is restricted by the final specifications. While CAD systems employ basic useful parameters to allow users to build the final design, GA utilizes preliminary designs from the beginning. CAD systems use primitives (points, lines and splines), which are controlled by users to build the design. In an evolutionary design system, it is GA that must modify designs to reach the final solution. When GA reach the solution, the design meets the final specifications. For this reason, the representation of an evolutionary design system based on GA must have a good parameter definition. Compared to the configuration design, a preliminary design is more difficult to computerize given its more marked emphasis on creativity. Therefore, the first step is to identify the ways to computerize the process involved in design. A bibliographic review sets the basis of using GA in the industrial design process.

Keywords: Industrial design · Optimization · Genetic algorithm · Computer-aided design

1 Introduction

Nature offers solutions to engineering problems. Nowadays, many engineering solutions mimic those that nature has built through evolution [1]. Evolution is the system that nature uses to adapt to its environment. It is a complex system, but has phases and processes that can be emulated with computation [2]. Genetic evolution is the most widely used tool by nature. Today we better know the processes that control genetics [3], and to the extent that we can emulate it with computational algorithms.

Genetic algorithms (GAs) emulate the process that characterizes genetic evolution using genetic operators [4]. These operators apply to a population from one generation to another. Each generation is evaluated according to the fitness function [5]. In this way, the evolutionary system perfects the solution in each generation.

F. Cavas-Martínez et al. (Eds.): *Advances on Mechanics, Design Engineering and Manufacturing II*, LNME, pp. 336–346, 2019.
https://doi.org/10.1007/978-3-030-12346-8_33

The engineering design process can have different levels of complexity. In industrial design, the desired solution optimizes quality values. However, the design process can involve many parameters that complicate optimization, and the design can contain many non-linear parameters. When the goal function has many local extremes, problems can appear [6]. In these cases, GA can solve the problem [7].

The most widely used optimization methods have problems [8]. The gradient method optimizes only local extremes [9], and the stochastic search method only disturbs the solution of the gradient method [10]. However, GA keep and evolve solutions in a search space.

GAs have been used in many optimization applications. We can see the application of GAs as: image processing [11], numerical optimization [12], programming [13], machine learning [14] or data structures [15]. This optimization method evolves solutions in a search space to obtain the desired result. Using GA as computer tools was introduced by Fraiser [16].

In this bibliographic review, we introduce a GA framework, and then see the relationship between GA and design. Next we see some examples of applying GA in industrial design.

2 Genetic Algorithms: Framework

A GA algorithm contains a search space formed by a population of individuals. Each subject, the so-called specimen, is composed of chromosomes. The so-called genotype forms the information contained in chromosomes. The so-called phenotype is the chromosome appearance. For each subject, the fitness function associates and evaluates an aptitude. From one generation to another, the population is submitted to the so-called genetic operators: crossing, mutation and cloning. Therefore, this process is evolutionary multiobjective optimization (EMO). We can establish a relationship between elements of nature and computation (Table 1).

Table 1. Relationship between nature and evolutionary computation

Nature	Computation
Specimen	Individual solution
Population	Set of solutions
Fitness	Evaluation
Crossover	Binary operator
Mutation	Unitary operator
Cloning	Recover/save operator

GA have the following stages (Fig. 1). First, the initial population is set up. In stage 2, the fitness function evaluates each specimen. If the evaluation is positive, the subject moves to the next generation through cloning. If the evaluation is negative, the subject is selected according to its level of adaptation. In stage 3, genetic operators create new specimens to form the next generation. Stage 2 is repeated until the satisfaction criterion is met.

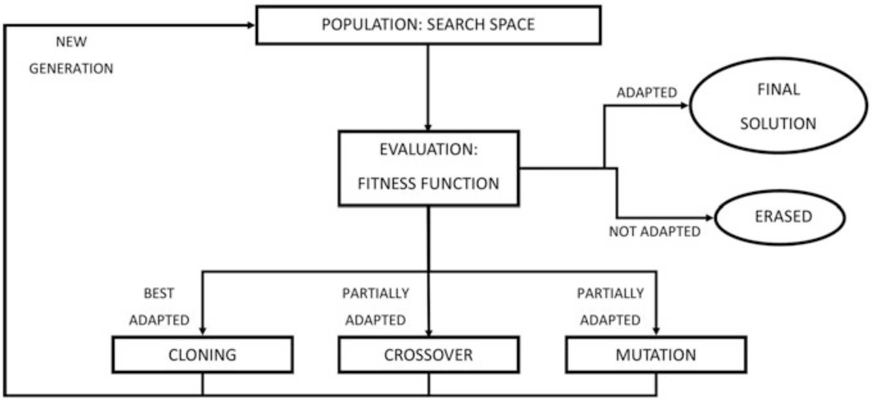

Fig. 1. GA schema

Each specimen represents a solution that needs to evolve from the beginning to reach the final specifications. To build the solutions space, we can use a complex or a simple representation. When we work in design, two problems can appear [17]. On the one hand, if we use a simple representation, GA spends a lot of time searching irrelevant regions. If, on the other hand, chromosomes representation is complex, the next generation can evolve without reaching the point of equilibrium. Zhao et al. [18] has worked in the gene theory to simply the design process.

2.1 Genetic Operators

In each step from one generation to another, GA apply genetic operators. There are three genetic operators: crossing, mutation and cloning. GA performance depends on the choice of crossing and mutation operators [17]. Cloning operator is a save/recover operator. Cloning consists in copying the solutions that adapt well to the final specifications. The crossover operator creates a new specimen from two other individuals. To make the crossover, GA use one crossing point or more by recombining chromosomes to form a new genotype. A recent survey [19] suggests that the geometric crossover operator (or the topological crossover operator) is well-defined when the solution is defined with distance (geometrically). Mutation creates a new specimen by making changes in its genotype. Modifications consist in changing part of the genotype or adding/removing chromosomes.

2.2 Fitness Function

As with nature, the survival of a specimen depends on its adaptation to the environment. In GA, the fitness function evaluates the fitness of each specimen. This fitness function must contain the requirements and specifications desired in the final solution. In this way, the degree each specimen's adaptation represents its distance to the final solution. The fitness function is considered the search core of GA. In design optimization, the fitness function is considerably important. An experimental study [20] establishes that an improved fitness function (IGA) has better convergence than traditional GA.

3 Involving GA in Design

Design generates solutions to cover products and specifications, and must contain information needed to manufacture them. Manufacturing imposes additional restrictions to design. Therefore, design is a task that involves a search to fulfill specifications and requirements. In 1962, a study by Asimov [21] showed the organization while solving a design task without taking into account the role that knowledge plays. It is a huge simplification, but the bases of this model are used in CAD-CAM systems.

Another model that enable knowledge of the design process is the so-called cognitive model [22]. This model can describe the designer's reasoning and decision making in design stages. However, this model is very difficult to translate into computational terms without using a broad simplification [23]. Here the design process is assimilated into parametric design thinking (PDT). PDT englobes the parametric design, the cognitive model and the computer-aided design. In short, the thinking process, involved in design, can be divided into two aspects: creative thinking and the pre-knowledge-based design (e.g. parametric design or configuration design [24]). We can see how GA can be used in these two contexts: creative design and configuration design.

3.1 Creative Design

Computerizing the creative design is a challenging task. In the creative design, there are two research lines: assisting human creativity and solving creative problems. In the former, users evaluate the population. So it is not necessary to define a fitness function. However, this causes a limited iteration speed because users cannot evaluate large populations. Frazer [25] concludes that a computer system´s creativity is attributed to the originator of the method.

A method to expand the search space is to reduce the number of the fitness function's restrictions. In this way, the system can change the initial representation. Then the system can build solutions beyond the optimization problem. The creative problem could have been solved by searching a solution that was not initially represented. This theory has been defended by Boden [26].

Another method is the so-called embryogenesis theory [27]. Here the fitness function evaluates the phenotype. Embryos are used to explore the search space. In this way, search parameters are not fixed.

Other studies have attempted to transform the creative process into an algorithm. Yang et al. [28] proposed dividing the search space into three: design flow space, knowledge inspiration space and solution operating space. In this way, GA acquire a certain level of creativity. Shieh et al. [29] propose three hybrid representations to obtain creative design alternatives. Golberg David [30] explains how to design creative GAs based on human comprehension.

3.2 Configuration Design

While the creative design is based on creative thinking, we can see modeling methods based on the problem's pre-knowledge. The configuration design [31] is based on the selection and assembly of components to build the final solution. The relationship between specifications and configurations is known. Therefore, it is a search to acquire an optimal configuration [32].

We can set a clear analogy between the configuration design and GA: configurable elements are encoded as genes with chromosomes. These elements evolve by GA (Fig. 2).

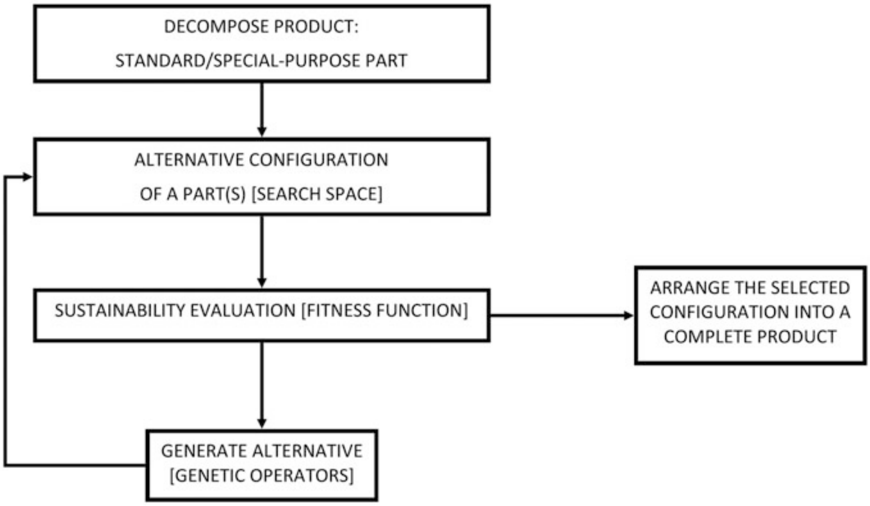

Fig. 2. Relationship between GA and the configuration design

In the configuration design, the configurable elements set the search space. The efficiency of GA is excellent due to the intensive exploring and exploitation that the genetic operators allow. Zou et al. [33] built the airspace grid design with configuration design-based GA. The search space, in this case, is formed by different grid configurations. GA evolve the grid testing alternatives configuration for each grid element. Da et al. [34] evaluated an evolutionary design composed of composite parts with different configurations. Andres-Perez et al. [35] used an enhanced mutation operator to generate different configurations in the search space. Chandrasekaran and Banerjee [36] resorted to a multiobjective evolutionary algorithm to offer an efficient and effective means to obtain a Pareto near-optimal set of solutions.

4 Applications of GA in Industrial Design

The potential use of GA in industrial design is remarkably high as they apply to most engineering fields: mechanical, electrical, aerospace and civil engineering. We analyze their application in three tools used in industrial design:

- Conceptual design
- Parametric design
- Reverse engineering

4.1 Conceptual Design

In conceptual design, either new components are used, or old components are combined in a novel way. There is no fixed method. Goldberg [37] presents a structure with four components: a problem to solve (the design challenge), someone to solve it (the designer), one conceptual design or more, and one method to compare alternative designs. This author presents GA as "a lower limit in the capacity of a designer in the processes of recombination and selection".

In conceptual design, GA are used as the method to compare and to search for alternative designs. This search can be guided by the designer, or the GA guide to the designer, to choose alternatives. Zhu et al. [38] used a GA-based selection method to accomplish an optimum multi-link transmission mechanism design. These authors adopted a polygon model into the iterative optimization process to describe the domination relationships of the individuals on the Pareto front. Mueller and Ochsendorf [39] proposed a computational approach for designing space exploration, which extends existing interactive evolutionary algorithms to enhance the inclusion of designer preferences. Zhang and Mueller [40] used GA to balance conflicting requirements in a conceptual structures design. Skiborowski et al. [41] introduced an interesting hybrid evolutionary-deterministic optimization approach for a conceptual design. They used successively relaxed mixed-integer nonlinear programming (SR-MINLP) to reduce the search time. Another approach to conceptual design optimization is shown by Zhang et al. [42], who followed an improved concurrent subspace optimization strategy (CSSO) and an improved differential evolution (DE) algorithm to solve the system-level and discipline-level optimization problems in conceptual design.

4.2 Parametric Design

In industry the shape of a component is important. Technical specifications and manufacturing costs depend on component shape (weight, torque, strength, etc.). In shape optimization, a parameter value must be found. The dependence function between the objective parameter with the shape variables can be non-linear or discontinuous. GA can assimilate this dependence function. Constraints can be determined by using a differential evolution algorithm in which the parameters are intermediate design variables [43].

The application of GA to parametric design has been successful in different studies. Gunpinar and Gunpinar [44] used GA to change the position of the points in parametric design. Particles are placed at points in the shape space by optimizing the fitness of the particle positions with a permutation GA (Fig. 3). Mostofizadeh et al. [45] performed the parametric design of a row of cylindrical film cooling holes to use GA in the optimization of gas turbines. In this case, the use of parameters improved the artificial neural network (ANN). Dandy et al. [46] compared the use of parameters in two GA: a traditional GA and a modified GA with improved operators. Renzi [47] utilized non-linear constraints to test individuals within each generation with geometric bounds and related parameters in an integrated design environment (IDe). Zhu et al. [38] compared the use of different model parameters in GA by choosing between the deterministic optimal design point and the robust optimal design point. Sekulski [48] selected the genotype by testing phenotype parameters.

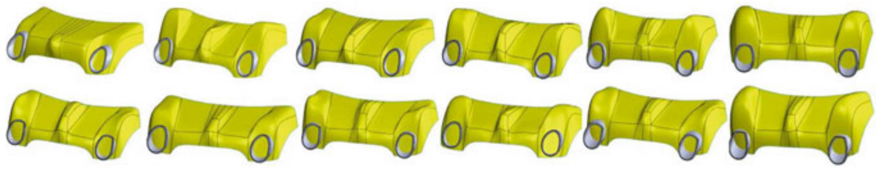

Fig. 3. Car hood models generated by parametric GA [44]

4.3 Reverse Engineering

A reverse engineering process begins by recording existing points by matching the model represented with the physically existing model. The next step is to establish a topology in the unstructured points cloud that reflects the mirror relations between the model and the real object. The rebuilding phase is based on belonging to subsets of the extracted elements.

For all these phases, GA have been investigated by different authors. Zhang et al. [49] applied Euclidean distance to duplicate the position of atoms in a material structure. Brunnstro and Stoddart [50] built mirror relations for points of correspondence in GA. The chromosomes of this model represent a correspondence between two sets of points (model point—real object point). The fitness function reflects the correspondence quality by calculating the distance between a pair of points. Its genetic code is composed of the transformation matrix parameters and the fitness function that minimizes the distance between each pair of points.

5 Results

The bibliographic review has introduced improvements into GA applications in industrial design. We describe the characteristics of different GA examples in several industrial design fields (Table 2). The design tool and case studies show that multiple GA applications are possible with different GA specifications.

We can see priority GA specifications to improve industrial design using GA. These parameters could be: a constrained fitness function, a chromosome substring, a

Table 2. Comparison between the GA applications reported in the literature

Reference	Application	Methodology	GA specs	Results
[39]	Rigid frame 2D model, generation of conceptual designs	Interactive GA	User-guided search space exploration	Lower computational cost than free exploration
[48]	Ship hull structural optimization (2D)	Nonlineal multiobjective optimization	Chromosome substring, constrained control parameters	~30% less generations needed to cover fitness
[38]	Multilink transmission concept design (2D)	Multi-objective optimization design (MOOD)	Ranked genetic operators	~35% total improvement
[47]	Integrated design environment (IDe) (3D modeling)	Integrated GA with numerical simulation	Constrained fitness function	98% iteration time reduction
[45]	Cooling holes CFD (3D)	Integrated artificial neural network-GA (ANN-GA)	Independent search space (database generation)	CFD Improvement
[44]	3D modeling	Particle tracing (PT) algorithm	GA optimizer of the particle position	72% iteration time reduction
[43]	Laminated plates (2D)	Differential evolution (DE)	Constrained genes modified objective reduction	Quick evolution
[49]	Reverse material engineering	GA with particle swarm optimization	Nondominated mutation	Promise tool for computer-driver material design

good selection of genetic operators. In [47], we predict that a constrained fitness function is better than a nondeterministic function. In [43] and [48], we observe that a constrained representation improves evolution. In [38], we discover that modified genetic operators are decisive to accomplish the final design specifications.

6 Conclusions and Future Work

GA can be successfully applied to industrial design. By even considering the creative factor in design, GAs can show the designer additional solutions.

In this bibliographic review, we study the definitions of the elements that compose a GA. We link these elements with the design process, which we see at both the creative and strictly procedural levels. Finally, we describe the characteristics of different GAs examples in several industrial design fields.

With all this, we see how GA have difficulties in creative and conceptual designs. However, we can verify how the fields of parametric design and configurations design have a high potential when applying GA. This is where our path is discovered by developing an optimization method in industrial design.

References

1. Kim SJ, Lee JH (2017) A study on metadata structure and recommenders of biological systems to support bio-inspired design. Eng Appl Artif Intell 57:16–37
2. Guizzo G, Vergilio SR (2018) A pattern-driven solution for designing multi-objective evolutionary algorithms. Nat Comput 1–14
3. Chaturvedi P, Kumar P (2015) Control parameters and mutation based variants of differential evolution algorithm. J Comput Methods Sci Eng 15(4): 783–800
4. Pavai G, Geetha TV (2018) New crossover operators using dominance and co-dominance principles for faster convergence of genetic algorithms. Soft Comput 1–26
5. Hanh LTM, Binh NT, Tung KT (2016) A novel fitness function of metaheuristic algorithms for test data generation for simulink models based on mutation analysis. J Syst Softw 120:17–30
6. Mirjalili S, Lewis A (2015) Novel performance metrics for robust multi-objective optimization algorithms. Swarm Evol Comput 21:1–23
7. Hamdy M, Nguyen AT, Hensen JLM (2016) A performance comparison of multi-objective optimization algorithms for solving nearly-zero-energy-building design problems. Energy Build 121:57–71
8. Qu X et al (2015) Intelligent optimization methods for the design of an overhead travelling crane. Chin J Mech Eng 28(1):187–196 (English Edition)
9. Keshavarzzadeh V, Meidani H, Tortorelli DA (2016) Gradient based design optimization under uncertainty via stochastic expansion methods. Comput Methods Appl Mech Eng 306:47–76
10. Jia G, Taflanidis AA, Beck JL (2015) Non-parametric stochastic subset optimization for design problems with reliability constraints. Struct Multi Optim 52(6):1185–1204
11. Sakthidasan K, Sankaran K, Nagappan NV (2016) Noise free image restoration using hybrid filter with adaptive genetic algorithm. Comput Electr Eng 54:382–392
12. Zang W et al (2018) A cloud model based DNA genetic algorithm for numerical optimization problems. Future Gener Comput Syst 81:465–477
13. Oliveira VPL et al Improved representation and genetic operators for linear genetic programming for automated program repair. Empirical Softw Eng 1–27
14. Wu CC et al (2018) A multi-machine order scheduling with learning using the genetic algorithm and particle swarm optimization. Comput J 61(1):14–31
15. Ting CK et al (2017) Genetic algorithm with a structure-based representation for genetic-fuzzy data mining. Soft Comput 21(11):2871–2882
16. Fraser AS (1957) Simulation of genetic systems by automatic digital computers I. Introduction. Aust J Biol Sci 10(4):484–491
17. Lin CD et al (2015) Using genetic algorithms to design experiments: a review. Q Reliab Eng Int 31(2):155–167

18. Zhao L et al (2016) A gene recombination method for machine tools design based on complex network. Int J Adv Manuf Technol 83(5–8):729–741
19. Pavai G, Geetha TV (2016) A survey on crossover operators. ACM Comput Surv 49(4)
20. Zhu Y, Cai X (2015) Convergence and calculation speed of genetic algorithm in structural engineering optimization. Metall Min Ind 7(8):259–263
21. Asimov M (1962) Introduction to design. Prentice-Hall, Englewood Cliffs, 135 pp
22. MacIntyre H (2015) A design model for cognitive engineering. Int J Technoethics 6(1):21–34
23. Oxman R (2017) Thinking difference: theories and models of parametric design thinking. Des Stud 52:4–39
24. Zhang T et al (2016) Intelligent fixture configuration design based on ontology and knowledge components. Jisuanji Jicheng Zhizao Xitong/Comput Integr Manuf Syst CIMS 22(5):1165–1178
25. Frazer J (2002) Creative design and the generative evolutionary paradigm. In: Creative evolutionary systems. Elsevier, pp 253–274
26. Boden MA (2004) The creative mind: myths and mechanisms. Psychology Press
27. Bentley PJ, Corne DW (2002) An introduction to creative evolutionary systems. In: Creative evolutionary systems. Elsevier, pp 1–75
28. Yang K et al (2016) A model for computer-aided creative design based on cognition and iteration. Proc Inst Mech Eng, Part C: J Mech Eng Sci 230(19):3470–3487
29. Shieh MD, Li Y, Yang CC (2018) Comparison of multi-objective evolutionary algorithms in hybrid Kansei engineering system for product form design. Adv Eng Inform 36:31–42
30. Goldberg David E (2002) The design of innovation, genetic algorithms and evolutionary computation. Kluwer Academic Publishers, USA
31. Levin MS (2016) Modular system design and evaluation, vol 373. Springer
32. McComb C, Cagan J, Kotovsky K (2017) Eliciting configuration design heuristics with hidden Markov models. In: International Conference on Engineering Design
33. Zou X et al (2016) Sectorization and configuration transition in airspace design. Math Probl Eng 2016
34. Da DC et al (2017) Concurrent topological design of composite structures and the underlying multi-phase materials. Comput Struct 179:1–14
35. Andrés-Pérez E et al (2016) Aerodynamic shape design by evolutionary optimization and support vector machines. Springer Tracts Mech Eng 1–24
36. Chandrasekaran S, Banerjee S (2016) Retrofit optimization for resilience enhancement of bridges under Multihazard scenario. J Struct Eng 142(8) (United States)
37. Goldberg DE (1991) Genetic algorithms as a computational theory of conceptual design. In: Applications of artificial intelligence in engineering, vol VI. Springer, pp 3–16
38. Zhu H et al (2016) Research on preference polyhedron model based evolutionary multiobjective optimization method for Multilink transmission mechanism conceptual design. Math Prob Eng 2016
39. Mueller CT, Ochsendorf JA (2015) Combining structural performance and designer preferences in evolutionary design space exploration. Autom Constr 52:70–82
40. Zhang Y, Mueller C (2017) Shear wall layout optimization for conceptual design of tall buildings. Eng Struct 140:225–240
41. Skiborowski M, Rautenberg M, Marquardt W (2015) A hybrid evolutionary-deterministic optimization approach for conceptual design. Ind Eng Chem Res 54(41):10054–10072
42. Zhang XB et al (2016) Multidisciplinary design optimization on conceptual design of aero-engine. Int J Turbo Jet Engines 33(2):195–208
43. Kameyama M, Arai M (2015) Optimal design of symmetrically laminated plates for damping characteristics using lamination parameters. Compos Struct 132:885–897

44. Gunpinar E, Gunpinar S (2018) A shape sampling technique via particle tracing for CAD models. Graph Models 96:11–29
45. Mostofizadeh AR, Adami M, Shahdad MH (2018) Multi-objective optimization of 3D film cooling configuration with thermal barrier coating in a high pressure vane based on CFD-ANN-GA loop. J Braz Soc Mech Sci Eng 40(4)
46. Dandy G, Wilkins A, Rohrlach A (2010) A methodology for comparing evolutionary algorithms for optimising water distribution systems. Water Distrib Syst Anal 2010:786–798
47. Renzi C (2016) A genetic algorithm-based integrated design environment for the preliminary design and optimization of aeronautical piston engine components. Int J Adv Manuf Technol 86(9–12):3365–3381
48. Sekulski Z (2014) Ship hull structural multiobjective optimization by evolutionary algorithm. J Ship Res 58(2):45–69
49. Zhang YY et al (2016) Inverse design of materials by multi-objective differential evolution. Comput Mater Sci 98:51–55
50. Brunnstrom K, Stoddart AJ (1996) Genetic algorithms for free-form surface matching. In: Proceedings of the 13th international conference on pattern recognition