Reconciling Form and Function through Generative Design

Nordin, Axel

2015

LUND UNIVERSITY

PO Box 117
221 00 Lund
+46 46-222 00 00

Reconciling Form and Function through Generative Design

Axel Nordin

LUND UNIVERSITY

DOCTORAL DISSERTATION

by due permission of the Faculty of Engineering LTH, Lund University, Sweden.

To be defended at Stora Hörsalen, IKDC, Sölvegatan 26, Lund. Date December 10, 2015 and time 09.15.

Faculty opponent

Professor Kristina Shea, ETH Zürich

Organization: LUND UNIVERSITY

Document name: DOCTORAL DISSERTATION

Date of issue: 2015-11-16

Author(s): Axel Nordin

Sponsoring organization

Title and subtitle: Reconciling Form and Function through Generative Design

Abstract: The current form-giving activity in industrial design is characterized by explorations that depend on the individual capability to mentally manipulate a solution space from which to select and express the intended result. Industrial designers frequently rely on artistic experimentation, aesthetic inspiration, or design briefs. These points of departure often result in satisfactory results, but they could be augmented by algorithmic form generation, optimization, and complex morphologies. By adopting this approach, the industrial designer would also be able to efficiently use forms that have previously been too complex to handle and evaluate manually, thereby gaining new ways of expanding his or her morphological repertoire. The difficulty of fulfilling these constraints is evident, as many iterations in the product development process are necessary between the different activities such as industrial design, engineering design, and production before a satisfactory design has been achieved. Additionally, if form is algorithmically generated and engineering and production demands are integrated in the process, partial or full transfer of the design activity to customers becomes a concrete option.

The possible results of achieving these goals are that designers gain a larger repertoire of morphologies to work from, the product development time can be reduced, the design concepts can stay true to the vision of the industrial designer, and the customers can tailor products to their needs. To achieve these goals, an approach is suggested that entails developing generative design tools that allow a user to design products with complex forms, while assisting them in ensuring the products' producibility and function. The focus is on increasing the integration between the industrial design activity and the other product development activities, more specifically engineering design and production

The purpose of this thesis is to investigate this approach, and to develop and test its technical feasibility and acceptance among designers and consumers. This has involved compiling an inventory of suitable morphologies, production systems and products, implementing and testing several generative design tools in industrial projects, looking into challenges concerning user manipulation of complex morphologies and industrial implementation, developing techniques for handling engineering constraints and objectives, testing the acceptance of the generative design tools with industrial designers and customers, and producing physical objects based on the output from the tools.

The results of this work show that this approach is feasible and is even already useful to the industry: the studies establish that there exists a feasible domain of application; they confirm that there are computational methods for handling engineering and user constraints and objectives; in the performed project the design process could be made to fit the needs of both consumers and designers; finally, physical products have been produced and have received acceptance by peers in international design fairs, thereby verifying and validating the output of the tools.

Keywords: Generative design, morphological repertoire, constraint handling, optimization, customization, industrial design, engineering design

Classification system and/or index terms (if any)

Supplementary bibliographical information

Language: English

ISSN and key title

ISBN 978-91-7623-540-9

Recipient's notes

Number of pages: xvi, 63

Price

Security classification

Reconciling Form and Function through Generative Design

Axel Nordin

Lund University

Division of Machine Design, Department of Design Sciences
Faculty of Engineering LTH, Lund University
P.O. Box 118
SE-221 00 LUND
Sweden

ISBN Print 978-91-7623-540-9
ISBN On-line 978-91-7623-541-6
ISRN LUTMDN/TMKT-15/1030-SE

Printed in Sweden by Media-Tryck, Lund University
Lund 2015

Acknowledgments

The research presented in this thesis has been carried out at the Division of Machine Design, Department of Design Sciences LTH, Lund University.

The research has in part been accomplished with support from Vinnova, which funded the hypothesis-testing project Renaissance 1.5, and from Fonden Innovativ Kultur, which funded the project Putting Nature to Work.

I would like to thank my main advisor, Professor Robert Bjärnemo, for his support and faith in the overall research idea. His experience and extensive network has been essential to anchor this research in industrial projects.

I am also grateful to my co-supervisor and co-author of several of the appended papers, Assistant Professor Damien Motte, for his comprehensive guidance and many stimulating discussions. His ideas and constructive comments on my work have been of utmost importance for the research.

I would also like to thank Andreas Hopf, lecturer at Industrial Design, whose original ideas concerning the morphological repertoire and natural-mathematical forms were the starting point for my research, and who has also contributed to several of the appended papers.

Thanks as well to Associate Professor Giorgos Nikoleris, who has contributed many ideas and insights regarding production, and the workshop staff for their guidance during the building of the prototypes presented in this thesis.

Lund, November 2015

Axel Nordin

Populärvetenskaplig sammanfattning

Industridesigners förlitar sig ofta på konstnärliga experiment, estetisk inspiration, eller så kallade designbriefer för att nå fram till lösningar på designproblem. Dessa utgångspunkter leder ofta till tillfredsställande resultat, men de skulle kunna kompletteras med verktyg hämtade från andra discipliner, till exempel algoritmer för att optimera en produkt, eller för att låta produktens form ”växa” fram som i naturen, så kallad generativ design. Genom att ta vara på möjligheterna som datorbaserade verktyg erbjuder så hade industridesignern alltså kunnat använda former som tidigare varit alltför komplexa för att hantera och utvärdera manuellt, och därmed få nya sätt att uttrycka sig på.

Avhandlingen fokuserar på hur man kan öka samspelet mellan industridesign och de andra aktiviteterna inom produktutveckling, som till exempel konstruktion och produktion genom användningen av generativa designverktyg, men även på hur liknande verktyg hade kunnat användas av konsumenter för att designa sina egna produkter. Detta har krävt en inventering av lämpliga former i naturen och matematiken, produktionssystem och produktkategorier. Ett antal generativa designverktyg har även utvecklats och testats i industrin och av konsumenter. En stor utmaning har varit hur användaren av ett generativt designverktyg på bästa sätt kan manipulera de komplexa former som genereras. En annan utmaning har varit hur tekniska krav på produkten, som hållfasthet eller tillverkbarhet, ska kunna uppfyllas när användaren inte själv har möjlighet att utvärdera dem.

Resultaten visar att det är möjligt att tillämpa generativ design inom produktdesign och att det redan är användbart för industrin: undersökningen visar att det finns ett stort område där det går att använda generativ design, den bekräftar att det finns beräkningsmetoder för att hantera tekniska och användarrelaterade krav och mål; den visar att det går att hitta användargränssnitt som passar både konsumenter och designers; och slutligen så har ett antal fysiska produkter tillverkats som har fått acceptans från såväl industri som konsumenter på internationella designmässor, vilket bekräftar att designsystemens resultat är giltiga och användbara.

Nyckelord: Generativ design, industridesign, konstruktion, datorbaserade metoder, optimering

Abstract

The current form-giving activity in industrial design is characterized by explorations that depend on the individual capability to mentally manipulate a solution space from which to select and express the intended result. Industrial designers frequently rely on artistic experimentation, aesthetic inspiration, or design briefs. These points of departure often result in satisfactory results, but they could be augmented by algorithmic form generation, optimization, and complex morphologies. By adopting this approach, the industrial designer would also be able to efficiently use forms that have previously been too complex to handle and evaluate manually, thereby gaining new ways of expanding his or her morphological repertoire. This development has been seen in other creative professions such as architecture, fine arts, modern music, and contemporary dance, but because of a different set of constraints linked to physical products, the progress has not been as rapid in industrial design. The difficulty of fulfilling these constraints is evident, as many iterations in the product development process are necessary between the different activities such as industrial design, engineering design, and production before a satisfactory design has been achieved. Additionally, if form is algorithmically generated and engineering and production demands are integrated in the process, partial or full transfer of the design activity to customers becomes a concrete option.

The possible results of achieving these goals are that designers gain a larger repertoire of morphologies to work from, the product development time can be reduced, the design concepts can stay true to the vision of the industrial designer, and the customers can tailor products to their needs.

To achieve these goals, an approach is suggested that entails developing generative design tools that allow a user to design products with complex forms, while assisting them in ensuring the products' producibility and function. The focus is on increasing the integration between the industrial design activity and the other product development activities, more specifically engineering design and production

The purpose of this thesis is to investigate this approach, and to develop and test its technical feasibility and acceptance among designers and consumers. This has involved compiling an inventory of suitable morphologies, production systems and products, implementing and testing several generative design tools in industrial projects, looking into challenges concerning user manipulation of complex morphologies and industrial implementation, developing techniques for handling engineering constraints and objectives, testing the acceptance of the generative design tools with industrial designers and customers, and producing physical objects based on the output from the tools.

The results of this work show that this approach is feasible and is even already useful to the industry: the studies establish that there exists a feasible domain of application; they confirm that there are computational methods for handling engineering and user constraints and objectives; in the performed project the design process could be made to fit the needs of both consumers and designers; finally, physical products have been produced and have received acceptance by peers in international design fairs, thereby verifying and validating the output of the tools.

Keywords: Generative design, morphological repertoire, constraint handling, optimization, customization, industrial design, engineering design

Appended Papers

This thesis includes the following appended publications:

Paper I
Nordin, A., Hopf, A., 2012, *Exploration of the Domain of Application of the Renaissance 2.0 Approach* (Technical report). Lund, Sweden: Division of Machine Design, Department of Design Sciences LTH, Lund University.

Paper II
Nordin, A., Hopf, A., Motte, D., Bjärnemo, R., Eckhardt, C.-C., 2011, "An approach to constraint-based and mass-customizable product design". *Journal of Computing and Information Science in Engineering*, *11*(011006).

Paper III
Nordin, A., Motte, D., Hopf, A., Bjärnemo, R., Eckhardt, C.-C., 2010, "Complex product form generation in industrial design: A bookshelf based on Voronoi diagrams". *4th Design Computing and Cognition Conference - DCC'10*, Stuttgart, July 12-14, 2010, pp. 701-720.

Paper IV
Nordin, A., Motte, D., Bjärnemo, R., 2013, "Strategies for consumer control of complex product forms in generative design systems". *39th Design Automation Conference - DETC/DAC'13*, Portland, OR, August 4-7, 2013.

Paper V
Nordin, A., Hopf, A., Motte, D., 2013, "Generative design systems for the industrial design of functional mass producible natural-mathematical forms". *5th International Congress of International Association of Societies of Design Research - IASDR'13*, Tokyo, August 26-30, 2013, pp. 2931-2941.

Paper VI
Nordin, A., Motte, D., Hopf, A., Bjärnemo, R., Eckhardt, C.-C., 2013, "Constraint-handling techniques for generative product design systems in the mass customization context". *Artificial Intelligence for Engineering Design, Analysis and Manufacturing - AI EDAM*, *27*(4), pp. 387-399.

Paper VII
Nordin, A., 2014, "Restart strategies for constraint handling in generative design systems". *40th Design Automation Conference - DETC/DAC'14*, Buffalo, NY, August 17-20, 2014.

Paper VIII
Nordin, A., 2015, “Challenges in the industrial implementation of generative design systems: An exploratory study”. *Artificial Intelligence for Engineering Design, Analysis and Manufacturing - AI EDAM.* Under review.

Contributions to the appended papers

Axel Nordin is the main author of all appended papers, and has also been responsible for the development of all generative design tools investigated in the appended papers, as well as the data collection. Damien Motte co-authored Papers II-VI. In Papers II, III and VI, he has provided background on the state-of-the-art of product development methodologies and design systems, as well as acting as a discussion partner and reviewer. He has also contributed to the experimental setup in Papers III, IV and VI. Andreas Hopf, together with Axel Nordin, has been responsible for the selection of suitable product typologies and production technologies, has given initial feedback on the design tools presented in Papers II-V, and authored sections on the current situation of industrial design presented in Papers II and III. Robert Bjärnemo is co-author of Papers II-IV and VI. He has provided input on product development methodology and helped review and structure the papers.

Also published by the author but not included in this thesis

Nordin, A., 2009, *Genetic algorithms and their applications within design* (In Swedish. Original title: *Genetiska algoritmer och deras tillämpningar inom konstruktion och design*), Master's Thesis. Division of Machine Design, Department of Design Sciences LTH, Lund University, Lund, 2009.

Hopf, A., Motte, D., Nordin, A., 2010, "Out of focus - The limits of a borderline profession", *25th Cumulus Conference*, Genk, 2010.

Bjärnemo*, R., Hopf*, A., Motte*, D., Nikoleris*, G., Nordin*, A., 2011, *Renaissance 1.5*. Vinnova, Stockholm.

Hopf*, A., Nordin*, A., 2011, *Putting Nature to Work*. Fonden Innovativ Kultur, Stockholm.

Motte*, D., Nordin*, A., Bjärnemo, R., 2011, "Study of the sequential constraint-handling technique for the evolutionary optimization of structural problems". *37th Design Automation Conference - DETC/DAC'11*, Washington, DC, August 28-31, 2011.

Nordin, A. 2012, *Transforming and Strengthening the Links between Industrial Design, Engineering Design and Production*. Licentiate Thesis. Division of Machine Design, Department of Design Sciences LTH, Lund University, Lund.

Nikoleris, A., Nordin, A., Palmqvist, H. & Sjöstrand, M., 2012, "Vem ska göra gruppindelningen och hur?". *7:e Pedagogiska inspirationskonferensen*, Lunds Tekniska Högskola, Lund, 2012.

* These authors contributed equally.

Table of Contents

1 Introduction

In this chapter, the background to the research problem is described, the research problem itself and the delimitations for the work presented in this thesis are defined, and, finally, the entire thesis is outlined.

1.1 General context

The traditional form-giving process in industrial design is characterized by explorations that depend on the individual capability of an industrial designer, or designer for short, to mentally manipulate a solution space from which to select and express the intended result. Designers frequently rely on artistic experimentation, aesthetic inspiration, or design briefs (Liu, 2003; Crilly, Moultrie, & Clarkson, 2004). These points of departure often lead to satisfactory results, but they could be augmented by the use of another form of creative approach, namely *generative design,* which, by computational means, supports the creation and evolution of the designs. By adopting *this approach*, the designer would also be able to efficiently use forms that have previously been too complex to handle and evaluate manually, thereby gaining new ways of expanding his or her morphological repertoire. The morphological repertoire can be defined as the infinite repository of all two- and three-dimensional forms, structures and compositions thereof (Hopf, 2009). Algorithmically controlled morphologies not only pave the way for the unimaginable, but also present methods to handle and adapt form to an intended purpose. In that sense, the form-giving process is augmented, or rather transformed into a form-finding process, concerned with the meta-design of rules and constraints from which desired or unintended, but feasible, results emerge.

However, in order for complex morphologies to be used within the industrial design discipline, several obstacles must be overcome. One problem is the multitude of constraints linked to the form-giving of artifacts: surfaces are often functional, the artifacts are produced in several copies – meaning that the product form must be modified to suit production systems – and cost control is consequently important.

Therefore, enhancing the designer's morphological repertoire is tightly linked to the possibility to generate technically feasible designs.

To meet this challenge, the overall goal of the thesis work presented here is to develop generative design tools that interactively help the designer to develop aesthetically and functionally interesting forms that fulfill the engineering constraints.

If this approach is successful, the benefits can go beyond enhanced creativity. First, in product development iterations are usually unavoidable between the designers on the one hand and the engineering design and production departments on the other hand. In effect, a concept proposed by the designer often has to be altered to fit engineering constraints. The modified concept is then sent back to the designer for further development, and so on. This limits the efficiency of the process, delays the time to market of a new or re-design product and, importantly, can also result in a final design that is only the shadow of what the designer originally intended. By taking into account the engineering constraints in the computer-based model of the product form, the designer would get a first-hand idea of potential problems, and would be able to modify the form until it fulfills both the engineering constraints and the designer's objectives and desires. This would result in *fewer iterations within product development projects and in a faster outcome*.

Second, if form is algorithmically generated and engineering and production constraints are integrated in the process, *partial or full transfer of the design activity to customers* becomes a concrete option. The demand for made-to-order products and services is increasing, even in markets where branding is important. "The world has changed. Consumers interact with brands on their own terms," says Trevor Edwards, Vice President, Brand and Category Management for Nike (Media, 2007). Many businesses have already implemented mass customization to some degree; see Section 3.2 for an overview. However, despite the increasing demand, there are very few major market players that *make customisation of the actual product form and structure* available to their customers. That approach has been adopted in other industries, for instance Threadless (Lakhani & Kanji, 2008) in the textile industry, or Innovate with Kraft Foods (Kraft Foods, 2006) in the food industry, but scarcely for engineered, discrete and physical products (furniture, electronic appliances, or other industrial goods). With the adopted approach this should become possible.

The customization of form in turn has several implications. Even if the traditional modes of product development remain dominant in the foreseeable future, the exploitation of niche markets becomes increasingly relevant and, more importantly, more profitable – the so-called "long tail theory" (Anderson, 2006). This becomes especially important in countries such as Sweden that cannot compete in production cost but can offer state-of-the-art flexible production facilities and a highly skilled workforce.

Finally, the act of personalizing a product creates a sense of product attachment with the customer as a result of the time and effort spent designing a unique and personal product, which might result in increased brand loyalty and longevity of the product, as the customer is less likely to dispose of a product to which he or she is attached (Mugge, 2007). This could be a way towards more sustainable products.

1.2 The approach

As described in the previous section, the purpose of the approach is to enhance the morphological repertoire of the designer by enabling the use of morphologies too complex to be manually handled. The approach proposed here to enable this augmentation of creativity is to develop generative design tools that integrate techniques and tools from engineering design and production with the industrial design activity. In practice, this means developing design tools that will enable designers with little or no technical knowledge to create designs that fulfill the engineering constraints of a product. By extension, this means that a design tool suitable for a designer without in-depth technical knowledge could be made available to customers directly, thus enabling them to configure the form and function of their products. As such a tool allows both designers and customers to design products, 'user' is employed in the text as a collective term to describe someone who designs the products using the design tool, for example a designer or customer. This can be an in-house designer who uses a tool to design a single object or a family of them. It can also be the customer (both consumers and businesses) or a sales person interacting with the customer.

The engineering constraints mentioned above are 1) functional constraints: the concept fulfills properly the intended technical function, must be reliable, takes into account physical laws; and 2) production constraints: the concept can be manufactured, assembled, packaged, transported, etc. For a piece of furniture like a table, the engineering constraints can be structural stability, ability to support a certain weight, and fitting the company's existing manufacturing equipment.

1.3 The Renaissance 2.0 research program

Andreas Hopf conceived the name Renaissance 2.0 in 2008. The expression was used to describe the widening of the morphological repertoire of the designer through the exploitation of forms inspired by nature and/or mathematics, thereby integrating art and science as during the Renaissance. Renaissance 2.0 is now the name of the research program of which this thesis is a part. The research program includes other areas of study, such as the necessary reflections upon the nature of the designer's role within the frame of this approach, and is extended as research progresses. Some of those domains are discussed in the conclusion, but are not included in the actual thesis work reported here.

1.4 Research problems

As mentioned in the previous section, the adopted approach should eventually **allow for an increased integration of industrial design into the product development process, and for an expansion of the morphological repertoire of the designer; this, in turn, would also result in significantly improved prospects for mass customization**.

This thesis aims at **investigating the technical feasibility and acceptance of this approach**. *Technical feasibility* means the investigations of the elements that are necessary to prove that generative design tools based on the approach *can* help fulfill the goals mentioned and focuses mainly on the "technical" aspects of the project, primarily those linked to the design tool. If the feasibility cannot be confirmed, then the project is not worth continuing. The *acceptance* refers to issues relating to how the approach is received by the users or the generative design tools, such as the designer's willingness or ability to move from designing products to designing algorithms, how generative design tools fit into companies' current product development process, or how consumers react to being given the opportunity of taking part in the design of their product-to-be. Many other steps are necessary to complete the research project. These are discussed in Future research (Section 5.2 below).

The aspects investigated in this thesis are presented below.

First it is necessary to verify that the "problem space" the approach can tackle is sufficiently large. Is this approach applicable to only a very limited set of products with a limited number of production systems and with only a few morphologies, or is it more widely applicable? In other words, is it worthwhile developing the approach at all? These issues are abridged in a first research question: **What is the domain of application of the approach?**

The core of the approach is the generative design tool. It must be possible for the user to evaluate and improve the performance of the designs in regard to objectives such as cost, weight and aesthetic appeal, without in-depth knowledge of how the parameters of the form affect the objectives. This can be achieved in a semi-automated manner through for instance interactive feedback, or through fully automated optimization algorithms. A fully automated design tool requires a fast and flexible optimization technique that generates good solutions reliably. The optimization technique must also be able to handle many different types of objectives and morphologies, without requiring substantial modification by a programmer. The problem of how the aesthetics of a form are assessed and optimized also arises, as this is not easily done computationally. Interactivity between the optimization technique and the user might be necessary. These issues are abridged in research question two: **How can a product form be optimized with respect to both qualitative and quantitative aspects?**

The issues regarding engineering constraints require special attention. The designs must satisfy constraints such as those linked to structural stability and manufacturing requirements, which are often hard to satisfy and time-consuming to evaluate computationally through simulations. Algorithms for constraint handling techniques must therefore be implemented, which, just like the optimization techniques, must be able to quickly and reliably generate diverse solutions to a wide range of problems, while requiring little input from a programmer. Much work has been done regarding optimization techniques, but pure constraint handling techniques for complex problems have not been the subject of as much research. It is therefore necessary to further build on the work previously done and possibly develop new techniques. These issues are summarized into research question three: **How can the engineering constraints be handled in such a way that diverse solutions can quickly and reliably be proposed to the user using a generic method?**

The interaction with the design tool must be efficient and intuitive for both designers and customers, enabling individuals with little or no knowledge of the constraints and objectives to create satisfactory designs. This requires establishing the optimal amount and type of control and how information regarding constraint satisfaction and objectives can be presented to the user in a responsive and efficient manner. This problem is not trivial, as users may have different preferences regarding the amount of control over the form, different educational backgrounds, and different motivations for using the design tool. The user might not be interested in or able to understand information about the engineering constraints, or have full control over the morphology. These issues are summarized into research question four: **What are the possible modes of interaction with the generative design tool for the user?**

In order for the approach to be successful, it needs to be both technically feasible and industrially accepted, that is, a generative design tool needs to be a good investment. Although generative design tools, which in theory are capable of handling any product type, are the subject of much research, most generative design systems need to be tailored to each industrial application. To be able to make any conclusions regarding the overall feasibility of the approach, it is therefore necessary to investigate the process of implementing generative design tools in industry. The tool development process is entangled in the overall product development process, and therefore offers additional challenges compared to developing a tool to solve an established research benchmark problem or a product with finalized specifications. Additionally, as the tool is part of the industrial design process, which heavily relies on the subjective judgement of the designer, the tool must be adapted to fit the workflow and direction of the designer. These questions are summarized in research question five: **What challenges are present in developing a generative design tool in an industrial project?**

1.5 Delimitations

To investigate the questions outlined in the previous section, a number of choices have been made based on the preliminary study of the extent of the Renaissance 2.0 framework (see Paper I). The choices have also been made by taking into account the typical challenges encountered in product development, such as integrating the design of product form with production and functional constraints, while optimizing some objectives such as cost or weight. In order to be able to investigate the main aspects of product development from the Renaissance 2.0 point of view within a limited time frame, the complexity of the morphologies studied, production technologies, constraint handling algorithms and products has been restricted, but it still illustrates the important problems and is by no means trivial.

In Paper IV, five 2D tessellations have been implemented, as this was determined to be a large enough pool of morphologies for the users of the application, and to provide differing levels of complexity in terms of manipulation and aesthetic appearance. In Papers V and VIII, three 3D morphologies have been implemented to ensure that the added level of complexity did not hinder the use or implementation of the generative design tools.

Five manufacturing technologies have been used during the validation efforts: computer numerically controlled (CNC) sheet metal bending, laser cutting, milling, slip-casting and metal casting. Combined, they enable the manufacture of many 3D designs from a range of materials. Rapid prototyping techniques such as 3D printing have been excluded, as restricting the application of an extended morphological repertoire to rapid prototyping would limit the approach to the types of products that are currently suitable for this fabrication technology. That is why "traditional" production systems such as laser cutting and CNC sheet metal bending have been chosen: they are proven technologies and are applicable to a larger number of materials. Moreover, many of Sweden's small and medium size manufacturing companies now rely on CNC machinery and could benefit from new applications for their equipment.

The products that were developed in 2.5D (used here to describe 3D objects based on an extruded 2D tessellation), rather than full 3D, are simpler in terms of the manufacturing process, their visualization in the design tool, and the level of complexity for the user, but they still require the same considerations as 3D objects in terms of engineering and production.

Three generative design tools for products in the furniture and lighting category have been implemented. Furniture provides a suitable test bed for innovation because any person understands what furniture represents, and it is still constrained in terms of weight, stiffness and visual appeal. The furniture sector is also an important part of Swedish industry, representing 20.8 billion SEK (2.5 billion USD as of April, Svenska trä- och möbelindustrin (TMF), 2015). Furniture is important also in terms of image in relation to what is widely known as Scandinavian design. Additionally, one product in the consumer electronics category and one in the industrial products category have been the basis for implemented design tools to provide insights into how more complex products affect the implementation and usability of the approach.

Since the investigated structures derived from mathematics or nature are highly non-linear and often discrete – as can be the constraints and objectives – a stochastic solver instead of a classical (e.g. gradient-based) optimization approach has been favored. A type of algorithm suitable for these types of problems is the genetic algorithm (Koza et al., 2004), although other stochastic algorithms such as simulated annealing are also applicable. The choice of using genetic algorithms instead of other non-continuous global stochastic search methods has been made because the results so far with using genetic algorithms are promising, the research community in this field is quite active, and many open-source libraries exist.

The interfaces have been chosen to be offline as hands-on instruction of the participants has been necessary and more information and feedback can be gathered by observation of the users while they are using the design tool. The implementation of an on-line interface that would include many more users has not been prioritized, as a first investigation showed that the main user interaction aspect that needed emphasis was the morphology handling.

In the generative design systems developed, the evaluation of subjective qualities of the product concepts generated by the design systems has been left entirely to the users. Other approaches are also possible, for instance through adaptive mechanisms that allow the system to learn the user's intent (see Wenli, 2008; Cluzel & Yannou, 2009) or through built-in rules that constrain the designs in terms of symmetry or proportions. While these approaches are promising, handing over the evaluation of subjective qualities to the generative design system could affect user acceptance, since that is the role traditionally played by designers and consumers. This aspect has therefore been left to future research.

As mentioned in Section 1.3, the impact of generative design systems on the roles of designers, consumers and the other members of the product development process is beyond the scope of this thesis, but is discussed in Future research (Section 5.2 below).

1.6 Outline of the thesis

Chapter 1 introduces the topic and the research problems at hand, as well as their limitations and delimitations.

Chapter 2 describes the underlying research process and the methods used.

Chapter 3 describes the frame of reference of this thesis.

Chapter 4 summarizes approaches and findings in the appended papers.

Chapter 5 contains conclusions about the research done so far and suggestions for future research.

2 Research approach

In this chapter, the research process is described, and the methods used for verification and validation are introduced.

2.1 Research process

The process has been iterative, and although each paper concentrates on different research stages, such as those described in the DRM framework by Blessing and Chakrabarti (2009, p. 15), there was originally no clear path from start to finish, and several of the research questions were discovered along the way. As for the overall motivation behind the research, the origin of the Renaissance 2.0 research program was the idea of finding a way for designers to utilize complex morphologies from nature and mathematics, thereby extending their morphological repertoire. Preliminary work had been done by Andreas Hopf to categorize morphologies from nature and mathematics depending on their origin and working principles (Hopf, 2009). When the author of this thesis was coupled to his work, this research project was extended to integrate engineering constraints that would allow the designer to be able to remain in control of the design for as long as possible. It soon became evident that the same design tools could be used by interested customers and could therefore take the concept of mass customization a step further. The additional benefits (opportunity for high-cost countries like Sweden, sustainability) ensued naturally. The research process has not been aimed at immediately implementing generative design tools that could achieve all of these goals at once, but rather to concentrate on individual aspects of generative design systems that needed to be tested to ensure the feasibility of the overall system.

The research process has been influenced by the order in which the research questions have been defined.

Regarding research question one *"What is the domain of application of the approach?"*: In order to answer positively to this question, it was necessary to identify a large list of products that could be designed using the approach to show that the domain of application for the approach is vast. To achieve this, the key elements required for implementing the approach first needed to be identified and then the members of each key element that together could form many products. The key elements were found to be morphologies, production technologies, materials and product typologies. Members of these key elements were then sought to ensure that a sufficiently large number of products could be designed through combinations of the element members. Preliminary work had been done by Andreas Hopf to categorize morphologies from nature and mathematics depending on their origin and rules of formation. This work was wide-ranging and published in (Hopf, 2009). This work was extended and the results are presented in Paper I. Additionally, all of the papers included in this thesis, except for Papers VI and VII, examine new morphologies, production methods and/or product typologies.

Regarding research question two *"How can a product form be optimized with respect to both qualitative and quantitative aspects?"*: A first version of the design tool described by the approach was devised together with a first case study. This was presented preliminarily in (A. Nordin, 2009) as part of a Master's thesis, and in its entirety in Paper II. Tables were chosen as the product typology, as they are low enough in complexity to facilitate rapid development of the design tool but still tied to a set of constraints and objectives that are common to all products. Based on the product typology selected, a survey of materials and production techniques was conducted to find those suitable for digital production of 2.5D structures. The requirement was that they could be fully automated in order to make customized products possible and economically feasible. CNC sheet metal bending, laser cutting, and adhesive joining were selected. In order to be able to handle engineering constraints and optimize the tables in terms of weight, a constraint handling and optimization technique based on genetic algorithms was developed. In summary, the results showed that it is in fact possible to use the proposed approach for designing products with complex form while fulfilling engineering constraints and optimizing objectives. However, a number of additional elements needed for verifying the feasibility of the approach were discovered. First, it is time-consuming to solve problems related to finding feasible product forms, they have a sparse solution space, and at the same time the users usually desire to choose among different solutions, highlighting the problem of diversity. All these elements were dependent on how the constraints were handled. It became necessary to show that there are efficient constraint-handling methods that take these elements into account. This became research question three. Secondly, the user interface developed and the amount of user influence it allowed over the product form was not satisfactory; it was obviously pointless to develop an approach where neither designers nor customers could handle the morphologies. This became question four.

A second case study was presented in Paper III where the approach was applied to a new product typology, shelves, and a new production method, CNC wood cutting. A new prototype of the generative design tool with different morphologies (2D tessellations) and several flat materials for the table problem was developed and presented in Paper IV. This confirmed that product form could be produced with respect to both user-related and technical objectives and constraints. Note that the first case study was made prior to the full investigation of question one in order to first have evidence of the overall relevance of the approach.

In Papers V and VIII, two additional product typologies, morphologies and production methods are introduced. The morphologies and products were chosen to demonstrate the approach using 3D morphologies and more complex products. The application in Paper V shows how 3D surface energy minimization can be used to create shapes for light reflection, and Paper VIII shows how a 3D morphology can be found together with a designer in an application relating to thermal dissipation in consumer electronics.

Regarding research question three *"How can the engineering constraints be handled in such a way that diverse solutions can quickly and reliably be proposed to the user using a generic method?"*: In Paper VI, the diversity, reliability and performance of three generic constraint handling methods was tested. The first method was developed in Paper II, and the other two, more established, techniques and two heuristics for configuring the constraint handling method were developed and tested in (Motte, Nordin, & Bjärnemo, 2011). The convergence time of the methods was investigated against an established benchmark, the ten-bar truss benchmark problem (Haug & Arora, 1979) in (Motte, Nordin, et al., 2011). Their reliability (convergence rate) and diversity were studied in Paper VI. Based on the results of Paper VI, another method for reducing the time needed to find solutions satisfying all constraints in problems where the convergence time was highly unpredictable was proposed based on restarting the search after a certain number of iterations in Paper VII.

Regarding research question four *"What are the possible modes of interaction with the generative design tool for the user?"*: A first interface allowing for a restrained control of the morphologies was developed and tested with five peer designers in Paper III. The results were not overwhelmingly positive, but they showed that the users seemed to split into two factions: one required full control over the design process, whereas the other one did not wish to delve into the details of the product's form but were rather satisfied with receiving product form suggestions from which they could choose. The study raised questions as to whether a tool with dual modes of control could lead to increased user satisfaction, and how to efficiently generate diverse design solutions for the user to choose from. As a result, a new user interface allowing two modes of control was developed and tested by a number of participants in Paper IV. Modes of interaction were also investigated in the industrial case studies in Paper VIII.

Finally, in order to verify the satisfaction of the structural and manufacturing constraints, and to gauge designers' and customers' acceptance of the approach, physical models were built using the intended manufacturing techniques and displayed at the Form/Design Center in Malmö, Sweden in 2010, the DMY design festival in Berlin 2010, the Stockholm Furniture Fair 2011, and the 100% Design fair in London 2011. The physical models also received media attention from the Swedish daily newspaper *Sydsvenskan* (Welin, 2010), the German *form* magazine (Fesser, 2010), and the SVT Science website (B. Nordin & TT, 2010). In general, the reception by people viewing the objects was positive, and many seemed interested in using such generative design tools.

Regarding research question five *"What challenges are present in developing a generative design tool in an industrial project?"*: In order to verify that the results from Papers I-VII were applicable in practice, and to investigate what challenges would be present when implementing a new design tool based on company requirements, case studies were conducted with two companies. The process and results are presented in Paper VIII. The case studies involved two industrial product development projects, one focusing on an engineering design application, and the other on an industrial design application. The challenges which were encountered were related to establishing the constraints and objectives that matched the experience of the engineering designers, and establishing the shape generation algorithm together with the industrial design team.

An overview of the research questions as addressed in each paper included in this thesis is shown in Figure 2-1 and Table 2-1.

Research question one “What is the domain of application of the approach?”

Conceptual framework

Hopf, 2009 P-I

Establishing key elements

P-I

Implementing key element members

2.5D P-II P-III P-IV A. Nordin, 2009

3D P-V P-VIII

Research question two “How can a product form be optimized with respect to both qualitative and quantitative aspects?”

Literature review

P-II P-III P-IV

Establishing fundamental structure of GDS

P-II P-VIII

Testing the feasibility

P-II P-III P-V P-IV P-VIII A. Nordin, 2009

Testing the acceptance

P-IV P-III P-VIII

Research question three “How can the engineering constraints be handled in such a way that diverse solutions can quickly and reliably be proposed to the user using a generic method?

Literature review

Motte, Nordin and Bjärnemo, 2011

P-VI P-VII

Establishing the requirements of CHTs

Motte, Nordin and Bjärnemo, 2011 P-II

Implementing CHTs

P-II P-VII

A. Nordin, 2009

Motte, Nordin and Bjärnemo, 2011

Comparing CHTs

Diversity P-VI P-VII

Convergence rate and time Motte, Nordin and Bjärnemo, 2011 P-VII

Research question four “What are the possible modes of interaction with the generative design tool for the user?

Establishing the need for different modes

P-III P-IV

Evaluating different levels of control

P-III P-IV P-VIII

Research question five “What challenges are present in developing a generative design tool in an industrial project?

Establishing the need for more research on industrial implementation of GDS

Investigation of implementation challenges

P-VIII

Figure 2-1. An overview of how each research question is being answered and in which papers

Table 2-1. An overview of the research questions being investigated in each paper

Research question/Paper	I	II	III	IV	V	VI	VII	VIII
What is the domain of application of the approach?	■	■	■	■	■			■
How can a product form be optimized with respect to both qualitative and quantitative aspects?		■	■	■				■
How can diverse solutions quickly and reliably be proposed to the user using a generic constraint handling method?		■				■	■	
What are the possible modes of interaction with the generative design tool for the user?			■	■				■
What challenges are present in developing a generative design tool in an industrial project?		■						■

2.2 Methods used

Several methods have been used to investigate and partially test the feasibility of the approach.

Qualitative interviews

To assess the first interface (Paper III), four professional industrial design peers tested the interface. They were asked to use the application to create and optimize one bookshelf per person and were questioned on the design tool's usability and their opinion of the amount or lack of control over the creation process and final solution. The interviews were recorded on video and transcribed. A qualitative analysis of the interviews was done as the number of available test subjects was limited and more information on key issues could better be found this way. The process is described in detail in Paper III. Interviews were also used as method for data collection in Paper VIII, where the formulations of constraints, objectives and shape generation algorithms were based on unstructured interviews.

User questionnaires

The interfaces developed later were tested on a somewhat larger scale than the partial control interface (Paper IV). The goals of the investigation were to determine whether these control modes were appreciated and whether one mode was significantly superior to the other.

The evaluation occurred at a design exhibition (at the Form/Design Center in Malmö, Sweden, see Figure 2-2) attracting visitors with an interest in interior and furniture design. The automated design generation system and two of the developed prototypes were presented. The visitors could test the system and design either a bookshelf or a table with the morphology of their choice. Seventeen of the visitors agreed to evaluate the handling mode of the diverse morphologies they had used. They were asked to estimate their level of satisfaction with the handling mode of each tested morphology on a visual analogue scale with scores ranging from 0 (not satisfied at all) to 1 (extremely satisfied). The participants were diverse in terms of computer experience and aesthetic training, as well as age and gender. The returned questionnaires were analyzed using a double-sided binomial test to establish whether or not a significant difference in satisfaction could be found between the no-control and full-control interfaces, and the qualitative comments were assembled. The process is described in detail in Paper IV.

Benchmarks

The constraint-handling techniques were compared using the well-known ten-bar truss benchmark (Haug & Arora, 1979) modified to be constrained in six ways in (Motte, Nordin, et al., 2011), and using the design system from Paper IV in Paper VI. Five different constraint-handling techniques were investigated, and two different constraint sequence selection heuristics were proposed. As the techniques all rely on a genetic algorithm, the results of the constraint-handling run are stochastic and thus must be analyzed using statistical methods. An ANOVA in conjunction with Tukey's test was used to determine whether the techniques and heuristics differed significantly in regard to convergence time, convergence rate, and diversity. The process is described in detail in Paper VI.

Case studies

The constraint handling and optimization systems have also been validated with case studies such as the generation of the tables and bookshelves. The process is described in detail in Papers II and III. Industrial case studies were the basis for Paper VIII where the challenges of industrial implementation of design tools were studied.

Figure 2-2. Setup for consumer testing of two of the interfaces at the Form Design Center in Malmö, Sweden

3 Frame of reference

In this chapter, the body of research upon which this thesis is based is reviewed. The first two sections elaborate on the context of the research project: integration of industrial design into the product development process, and the mass customization paradigm. Then background is given to the issue of control in the design process, related to research question four. The basics behind the structure of the developed generative design system (optimization and constraint handling) are introduced in Section 3.4, and finally a review of related generative design systems is presented in Section 3.5.

3.1 Integration of industrial design into the product development process

There are different schools of thought regarding the product development process; see for instance (Olsson, 1976; Pahl et al., 2007; Ulrich & Eppinger, 2012). However, the basic formulation of how the product development process is sequenced is roughly the same (Motte, Bjärnemo, & Yannou, 2011). The issue tackled here is how the industrial design activity is integrated into the product development process, specifically with the engineering design and production activities.

Industrial design has an important role in the development of products. According to Ulrich and Eppinger (2012), industrial design helps increase product appeal, customer satisfaction, willingness to pay, and sales. However, the involvement of industrial design can also increase the number of iterations necessary to fulfill both the desires of the industrial design department and the embodiment and production departments (Ulrich & Eppinger, 2012, p. 198). While it can increase the manufacturing cost of a product by introducing more exclusive materials or forms difficult to manufacture, Ulrich and Eppinger note that a well-integrated industrial design department can actually reduce production costs and improve product functionality. Clearly, the benefits of investment in design in consumer-oriented companies are by and large undisputed. Often, design is the decisive factor in the marketing mix to maintain and gain market share, or to enter into or create entirely new markets. What is often underexposed is that design is the central contact point in customer relationship management, because design is, above all, a means of communication (Bruce & Bessant, 2002, p. 19; Crilly et al., 2004).

The timing of the industrial design effort is dependent on the nature of the product developed. In market-driven products, the industrial design activity often takes place at the beginning of the product development process, as the design of the product is an essential attraction for customers. In technology-driven products the industrial design effort is often introduced at the end of the product development process to give the product a shell that communicates certain information and semantics to the user. Throughout the product development process of a user-driven product the emphasis on industrial design is much higher, as the user will often interact with the product, and thus require good ergonomics, maintainability, ease of use, and aesthetic appeal. An important step towards gaining the advantages of industrial design while cutting unnecessary iterations and cost increases is to better integrate it into the rest of the functions of product development, specifically engineering design and production. Ulrich and Eppinger (2012, pp. 217–220) go in that direction: they propose an industrial design process model of six steps: investigation of customer needs, conceptualization, preliminary refinement, further refinement and final concept selection, control drawings or models, and coordination with engineering, manufacturing and external vendors. In their model, however, the integration of industrial design with the other functions of product development is only realized in the last step, when a large amount of work has already been put into the product and many design parameters have been fixed.

Another approach that also aims to better integrate the elements of product development, partly through the use of computer-based tools, is concurrent engineering, which is defined as "a systematic approach to the integrated, concurrent design of products and their related processes, including manufacturing and support. This approach is intended to cause the developers from the very outset to consider all elements of the product life cycle, from conception to disposal, including cost, schedule, quality and user requirements" (Pennell & Winner, 1989, p. 648). By integrating critical engineering and production requirements early in the industrial design process, the likelihood for a designer to "get it right first time" – or to eliminate less severe ulterior changes – is higher. In the domains of product platform and product family design, the algorithms developed to automate the generation of product variants and models (e.g. Simpson, Siddique, & Jiao, 2006) take engineering and production constraints into account, but not aesthetics.

3.2 Mass customization in relation to product development and the economy

In the traditional product development process, one product is developed at a time, although product size ranges and modular products can be developed using a similar approach; see for instance (Pahl et al., 2007, Chapter 9). Developing size ranges for a product enables efficient use of development time and manufacturing resources to offer the customer a limited selection of products performing the same basic function, but with some differentiated features. The modular approach also enables the products to incorporate modules performing different functions without having to redesign the basic product framework; see (Ulrich & Eppinger, 2012) for an overview of modular product architecture. In essence, the modular approach allows for efficient improvement of functions, add-on functionality, adaptation to certain uses, replacement of worn out parts, and reuse of technology for launching new products. This is commonly seen in the automotive industry where product platforms enable many different car models to share common functionalities such as the drive train or chassis. Another advantage of a modular product is the possibility to increase the variability of the product. For example, many of the Swatch watch models rely on the same product platform but are differentiated by combining diverse models of hands, faces, and bracelets (Franke & Piller, 2004).

Taking advantage of the possibilities offered by the variability and modular design, the concept of mass customization was introduced by Davis (1987), and further developed in (Pine, 1993). Kaplan and Haenlein (2006, pp. 176–177) defined it as "a strategy that creates value by some form of company-customer interaction at the fabrication and assembly stage of the operations level to create customized products with production cost and monetary price similar to those of mass produced products". Although the definition includes any sort of customization, including customization during the manufacturing phase, in reality most mass customized products today are purely superficial in their modifiability, and only the assembly phase is different from ordinary mass production. Several examples of this are given below.

Mass customisation and its representation in online product configurators is a relatively common tool to boost sales, brand awareness and brand loyalty. Studies of consumers' willingness to pay in relation to the possibility to customize a watch has shown that, on average, a 100% increase in the willingess to pay could be measured for watches that could be customized compared to non-customizable watches (Franke & Piller, 2004).

These online product configuration websites are offering many diverse forms of mass customization, a large bandwidth of customisation options, navigation techniques and visual quality. Screenshots of the customization process are shown for NIKEiD, adaptare.ch, and Bike by Me in Figure 3-1, Figure 3-2, and Figure 3-3, respectively. Most common are changes of colors and graphics, the display of the customised product realised through playback of pre-recorded 2D or 3D images.

Despite the steady upsurge in online product configuration, there is no major market player that makes customisation of the actual product form and structure available to its customers. It should also be noted that none of these configurators include evaluation of manufacturing or structural constraints. In (Mugge, 2007) it is also noted that there were no configurators available that enabled the consumer to become mentally deeply involved with the design process.

The inception of mass customization can be linked to a change in customer behavior and preferences. The proliferation of independent fashion labels, co-working spaces or car-sharing initiatives indicates that people do want to be empowered commercial actors – but on their own terms (Friebe & Ramge, 2008). Online discussion forums, blogs and help websites like WikiAnswers (http://wiki.answers.com/) often surpass commercial product assistance and service offerings. Product subversion and design-hacking blogs like instructables (http://www.instructables.com/), There, I Fixed It (http://thereifixedit.com/) or IKEA Hackers (http://ikeahacker.blogspot.com/) reveal an interpretative wit and ingenuity sometimes excelling that of professional designers. Social media like Flickr (http://www.flickr.com/), iStockphoto (http://www.istockphoto.com/) or Stock.XCHNG (http://www.sxc.hu/) have democratized the photographic image trade, once the realm of highly profitable stock-photo agencies. Social commerce platforms such as Etsy (http://www.etsy.com/) or DaWanda (http://www.dawanda.com/) showcase self-made products of low complexity, ranging from amateurish bricolage to astonishingly well-designed objects. Channeled through careful brand monitoring, any coder can develop and sell iPhone compliant applications under iOS Developer Program (http://developer.apple.com/iphone/program/). And more importantly, online fabrication providers like Fluid Forms (http://www.fluid-forms.com/) or Freedom Of Creation (http://www.freedomofcreation.com/) put rudimentary means of production into the public domain. It is these examples that foreshadow a new economic paradigm, that is to say that the longing for participation and personally motivated entrepreneurship is turning consumers into prosumers (Toffler, 1971) and micropreneurs – in a world of countless niches. The future success of commercial and non-profit endeavors will no longer be determined by finding ways to make people do what enterprises want, it will depend on letting them do what they like (Reynolds, 2006).

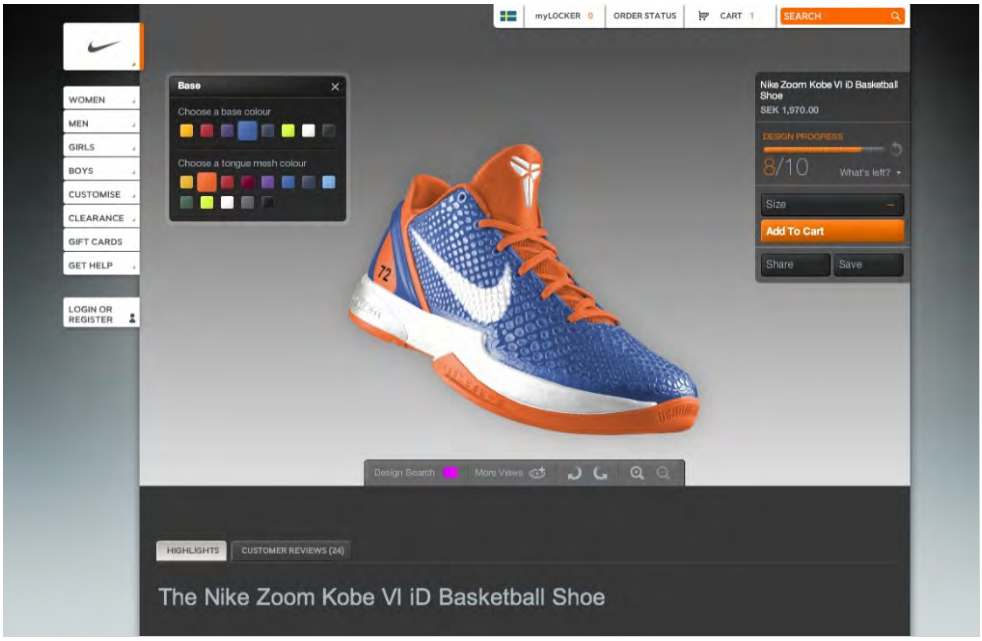


Figure 3-1. Customization of a basketball shoe at NIKEiD (http://www.nikeid.com/)

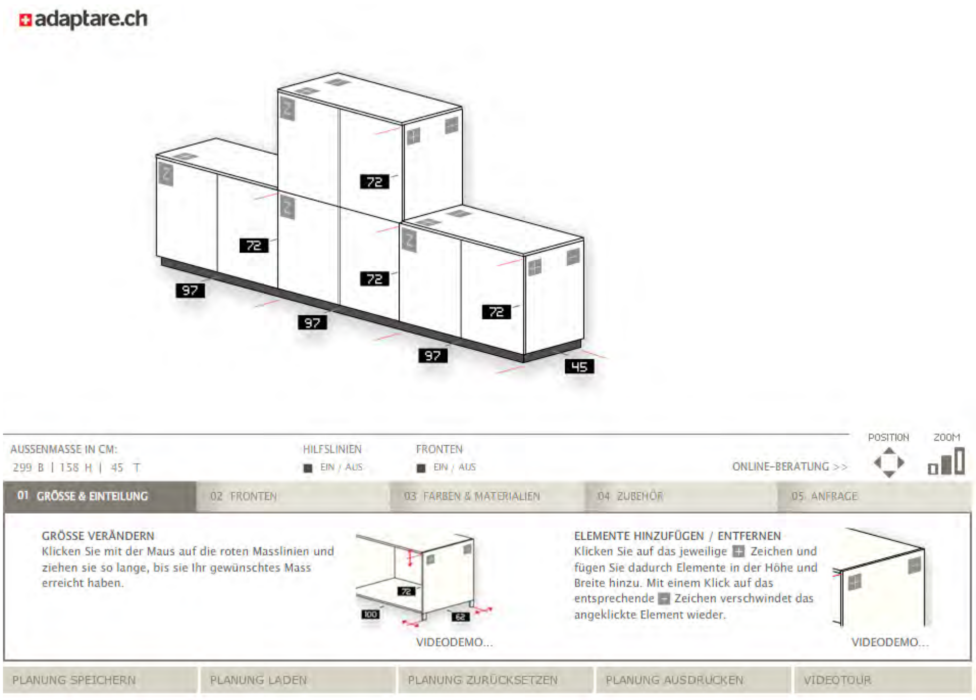


Figure 3-2. Shelving customization interface of adaptare.ch (http://adaptare.ch/)

Figure 3-3. Bicycle customization interface of Bike by Me (http://www.bikebyme.com)

3.3 The issue of control in the design process

The initial question regarding functionality, interactivity and output of a generative design tool is whether the designer or customer is willing to relinquish control to a certain degree. A "romantic" designer personality may not accept that algorithms seemingly restrict creativity; an experimentally open-minded designer personality may actively look for emergent behavior to find unexpected solutions. In case the designer or customer is attracted to generative design methodologies, the question then is in what way the degree of freedom is limited – and for what reason. The resultant output may either only remotely resemble the chosen startup design, or turn out to be fairly predictable. One could therefore speak of controlled serendipity, wherein the number of constraints – whether aesthetic or functional – is the determining factor.

The sheer number of constraints – determining the degree of usability – must also be considered. Whereas a skilled designer may wish to adapt a generative design and optimization application with constraints beyond the aesthetic, for example, complex functional and production constraints, an unskilled consumer is likely to shrink from venturing much beyond the aesthetic and overall dimensioning. Therefore, such an application, if generalized for the widest possible range of input morphologies and output product typologies, needs a very customizable graphical user interface (GUI) in order to show and hide complexity depending on the task at hand.

More specifically, regarding mass customization, Mugge (2007) finds that allowing consumers to personalize their products in some manner increases their attachment to the products, increases the perceived value of the product, and as noted in Section 3.2 it can increase the willingness to pay by as much as 100%. However, the problem remains in which way and to what extent the products should be customizable. Mugge notes that there are a number of dimensions to customization, ranging from physical modification of the products to selection from a large number of pre-defined versions. The dimensions involve mental effort, physical effort, flexibility of the design system, to what extent the customer is forced to customize the product, whether the product is customizable in terms of aesthetics or functionality, when the personalization is performed (before purchase, before use, or during use), and to what extent the personalization is deliberate or a result of, for instance, wear and aging with use. The framework suggested in this thesis is concentrated on the deliberate customization before purchase using generative design tools. The question of the amount of control over the customization and the mental effort required from the customer is part of research question four.

Piasecki and Hanna (2010) address the issue of control in relation to complexity and customer satisfaction. They build on the research done by Schwartz (2004), who defined “the paradox of choice”: a large amount of choice in such products is associated with happiness and satisfaction, but too much choice can lead to dissatisfaction and confusion. Indeed, Piasecki and Hanna showed using a configurable object that there exists an optimal amount of control, but that it is not only the amount of choice but also the amount of *meaningful* choice that influences customer satisfaction. They also introduced a method for handling complex choices using an interactive genetic algorithm that lets customers quickly steer the search for a design that satisfies their needs and wishes (see Kelly, 2008 for an in-depth explaination of interactive genetic algorithms). A similar approach can be seen in (Kelly, Wakefield, & Papalambros, 2011) that uses the interactive genetic algorithm to find consumer preferences regarding bottle shapes. The user is shown various designs of a parametrically defined bottle shape and is asked to select the most appealing version. The input is then used to create a new generation of bottle shapes, which in turn is evaluated by the user once more. This cycle is iterated until the user is satisfied with the design. The approach is shown to quickly lead to pleasing results while limiting the complexity faced by the user. No constraints regarding manufacturing or structural integrity are included in the evaluation, however, and in fact the study shows that user preferences are not based on the functional aspects of the bottles but rather on their aesthetics. This emphasizes the importance of implementing an effective method for the design tool to handle the engineering constraints when utilizing an interactive genetic algorithm. This is the topic for research question three.

3.4 Optimization and constraint handling systems

In the general case, some functional, aesthetic, engineering and manufacturing objectives $\mathbf{F(x)}$ need to be minimized or maximized (for example costs and weight), while other functional, aesthetic, engineering and manufacturing conditions are constraints $\mathbf{G(x)} \leq 0$ and $\mathbf{H(x)}=0$. This is a multi-objective optimization problem:

$$\begin{aligned} &\textit{minimize} \;\; \mathbf{F}(\mathbf{x}) \\ &\textit{subject to} \; \mathbf{G}(\mathbf{x}) \leq \mathbf{0} \; \textit{and} \; \mathbf{H}(\mathbf{x}) = \mathbf{0} \end{aligned} \tag{3.1}$$

The field of optimization can be traced back to Gauss, who formulated the first mathematical optimization technique, the steepest descent method. During the research-intense World War II, much progress was made in the optimization field to plan optimal strategies, notably the introduction of linear programming made by Leonid Kantorovich in 1939 (Kantorovich, 1940), and the simplex method used to solve linear programming problems created by George Dantzig in the 1940s (Dantzig, 1951). The field then quickly evolved into what it is today; the first description of an evolutionary algorithm was made in 1954 by Nils Aall Barricelli at the Institute for Advanced Study in Princeton (Barricelli, 1954). Evolutionary techniques became more common during the 1960s, and in 1970 the Australian quantitative geneticist Alex Fraser described all of the essential elements of modern genetic algorithms (Fraser & Burnell, 1970). The genetic algorithm in particular was detailed in the work of John Holland in the early 1970s, and particularly in his book *Adaptation in Natural and Artificial Systems* (Holland, 1975). Although so far largely theoretical in their use, evolutionary algorithms became the focus of commercialization in the late 1980s when General Electric started selling a mainframe-based toolkit based on evolutionary optimization designed for industrial processes.

As mentioned in Section 1.5, since the structures derived from mathematics or nature that are investigated in this research project are highly non-linear and often discrete – as can be the constraints and objectives – a stochastic solver instead of a classical (e.g. gradient-based) optimization approach is to be favored. A type of algorithm suitable for these types of problems is the genetic algorithm (Koza et al., 2004). A genetic algorithm tries to artificially simulate the process of evolution (Holland, 1975) by which the structures in nature were first created. For a review of applications using genetic algorithms for multi-objective optimization see Coello Coello (2000).

The handling of the constraints deserves special attention. Some constraints are hard to fulfill. In the table application for instance, few of the generated solutions fulfilled the manufacturing constraints during the first half of the search (see Paper II). Other constraints are time consuming to fulfill. For the analysis related to structural problems, finite elements techniques may be required for instance, which may take hours to complete for a single design. Moreover, it is necessary that the system has a high probability to quickly converge on one or many solutions so as not to waste customers' time and company resources. Finally, it is necessary to give the user solutions with a high level of diversity to increase the likelihood that he or she will find one that is satisfactory and to give the user a sense of control. These challenges require substantial research effort, and although small in comparison with the sum of works on evolutionary computing, the number of publications dedicated to constraint handling techniques is increasing at a rapid pace. A website gathering studies in that area lists more than 870 references (Coello Coello, n.d.). These techniques can be classified in four or five categories. These different categories are reviewed in (Motte, Nordin, et al., 2011); for an overview, see (Michalewicz & Schoenauer, 1996; Coello Coello, 2002; Mezura-Montes, 2004; Yeniay, 2005). Importantly, it can be noted that many of the constraint handling methods reviewed are specialized to certain types of problems or require substantial fine-tuning of algorithm parameters to work well. Only a few techniques generalize well, among them the sequential constraint handling techniques and weighted sum technique. The sequential constraint handling techniques handle the constraints in sequence instead of handling all constraints simultaneously, thus reducing the time needed per iteration, but potentially increasing the number of iterations needed to converge on a solution if not sequenced properly. Heuristics for choosing a good sequence are therefore investigated in (Motte, Nordin, et al., 2011).

3.5 Generative design systems

The available technology has always shaped product design, architecture and art. Advances in material and manufacturing technology partially shaped the architecture and products of the industrial revolution, and later, the introduction of for instance Bakelite and its many interesting properties played an important role for product design. Today, computer-controlled production technologies coupled to computer-aided design tools (CAD) are changing the aesthetics of architecture, and to some extent product design. When introduced in the 1960s, CAD tools were used in aircraft and automotive industries to handle parametric curves and surfaces, and to control CNC machines developed during the same period (Farin, 1993). CAD tools were soon extended to also perform tasks such as drawing generation and plotting, which had previously been very time-consuming. Although they were originally created to automate and improve the quality of the previously manual tasks, CAD tools soon began to change the way products were being designed. The parametric formulation of form coupled to 3D visualization of form enabled quick experimentation and variant creation, which had previously been next to impossible. Complex shapes with smooth curves became commonplace in the 1990s when CAD tools such as Dassault Systèmes Catia and PTC's ProEngineer enabled designers to easily define such forms, and a similar evolution could be seen in architecture with works such as Frank Gehry's Guggenheim Museum in Bilbao, which was aided by software like Catia. Recently, through new tools such as McNeel Rhinoceros in conjunction with the graphical algorithm editor Grasshopper, the open source Processing language, and Dassault Generative Systems, architects, and to some extent product designers, have been given an easy way to script these parametric tools, which has initiated the use of complex generative architecture such as Herzog & de Meuron's Beijing National Stadium. The designers are no longer changing the parameters and forms directly, but rather changing the algorithms that rule them. This enables a fast exploration of forms based on some underlying logic. The generation of the forms and variation of the parameters can in turn be controlled either manually, by search algorithms, or even by serendipity.

Generative design systems that take into account engineering constraints as well as aesthetic intent have existed in the field of architecture since the 1970s, see, for example, (Frazer, 2002) and (Bentley & Corne, 2002), but have rather been the object of isolated research studies in industrial design. The following survey of the generative design systems in industrial design has in part been used in Paper VI.

Engineering constraints have, among other fields, been studied within engineering design. For example, Agarwal and colleagues (Agarwal & Cagan, 1998; Agarwal, Cagan, & Constantine, 1999) have developed a coffee maker shape grammar associated with parametric cost to develop new products (see Cagan, 2001 for a review on the use of shape grammars in engineering design).

In industrial design, generative design has primarily been used for stylistic purposes. In the seminal work of Knight (1980), a parametric shape grammar was developed for the generation of Hepplewhite-style chair backs. Orsborn et al. (2006) employed a shape grammar to define the boundaries between different automotive vehicle typologies. Recent works have focused on branding related issues. With the help of shape grammars, new designs based on the Buick (McCormack, Cagan, & Vogel, 2004), Harley-Davidson (Pugliese & Cagan, 2002), Coca-Cola and Head & Shoulders (Chau et al., 2004) brands were developed. Further research is undertaken towards rules that are linking form and brand in, for instance, (Cluzel & Yannou, 2009) for genetic algorithm-based systems and (Orsborn, Cagan, & Boatwright, 2008) for shape grammars. Federico Weber developed the Xylem table project

(Weber, n.d.) (see Figure 3-4) which enables a user to design a table based on the Voronoi diagram by adding and moving Voronoi points. The system calculates the shape of the table and the support structure and is able to output drawings for manufacture. The system does not take into account manufacturing and structural constraints, however.

Some works are crossing the boundaries between engineering and industrial design, taking into account engineering constraints and aesthetics. Shea and Cagan (Shea & Cagan, 1999) used a combination of shape grammar and simulated annealing for both functional and aesthetic purposes and applied it for truss structures (truss structures are commonly used for both industrial applications and consumer products). Shape grammars were used to generate new designs, while the simulated annealing technique directed the generation towards an optimum. The design objectives were functional (minimize weight, enclosure space and surface area), economic and aesthetic (minimize variations between lengths in order to get uniformity, and make the proportions respect the golden ratio). Their model has been re-used in (Lee & Tang, 2006) and (Lee & Tang, 2009) (shape grammar and genetic algorithm) to develop stylistically consistent forms, and has been applied to the design of a camera. The designs generated took into account the constraints linked to the spatial component configuration. A designer was in charge of the aesthetic evaluation, following the interactive genetic algorithm paradigm. Ang et al. (2006), also using shape grammars and genetic algorithms, developed the Coca-Cola bottle example of (Chau et al., 2004), and added functional considerations (the volume of the bottle) that were constrained to approach the classic Coca-Cola bottle shape. Morel et al. (2005), within the interactive genetic algorithm paradigm, developed a set of chairs optimized for weight and stiffness. Finally, Wenli (2008) developed a system that, through adaptive mechanisms, allows it to learn the designer's intent faster; that system was implemented as a plug-in for a CAD system and applied to boat hull design. KRAM/WEISSHAAR developed the Breeding Tables project (see Figure 3-5), which generates variations of a table design using a genetic algorithm that modifies a set of parameters ruling the support structure (Kram & Weisshaar, 2003). The system does not take stability into account, but it does ensure the producibility of the designs through constraints on the parameters. The Computational Chair project developed by EZCT Architecture & Design Research (2004) (see Figure 3-6) also uses genetic algorithms to generate design variations of a chair built from pieces of plywood glued together, but the algorithm in this case also minimizes the weight and ensures the structural stability of the chair through finite element analysis.

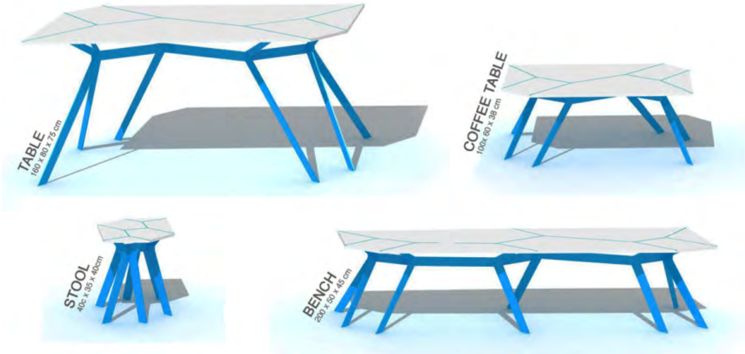


Figure 3-4. Variations of the Xylem table project by Federico Weber (©Federico Weber, http://federicoweber.com/)

Figure 3-5. Two variations of the KRAM/WEISSHAAR Breeding Tables project (©KRAM/WEISSHAAR, http://www.kramweisshaar.com/)

Figure 3-6. One variation of the Computational Chair project by EZCT Architecture & Design Research (2004) (©EZCT Architecture & Design Research)

4 Summaries of the appended papers

4.1 Paper I – Exploration of the Domain of Application of the Renaissance 2.0 Approach

The purpose of this report was to show that the domain of application of the Renaissance 2.0 approach is large. In order to do this, one must first identify the key elements of the approach, and then find suitable members of each key element that in combination enable the design of a large set of products.

The first part of the study resulted in a list of necessary elements; the key technical elements were found to be product typologies, morphologies, production technologies, and materials. These elements as well as their interdependencies are shown in Figure 4-1.

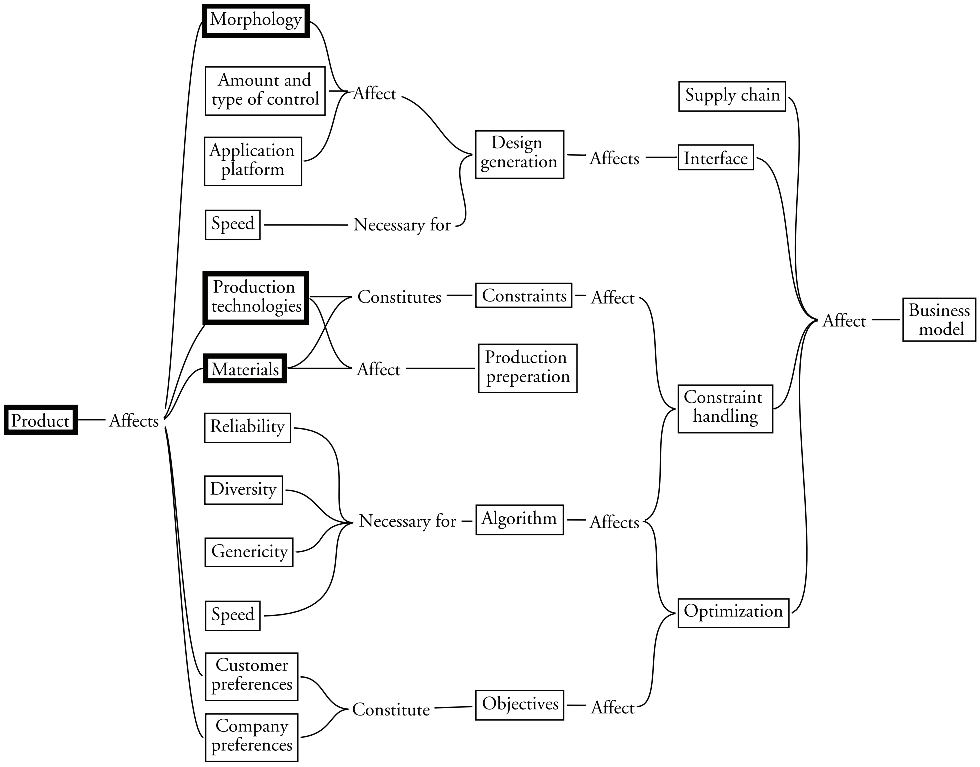


Figure 4-1. A network of interdependencies of a generative design system

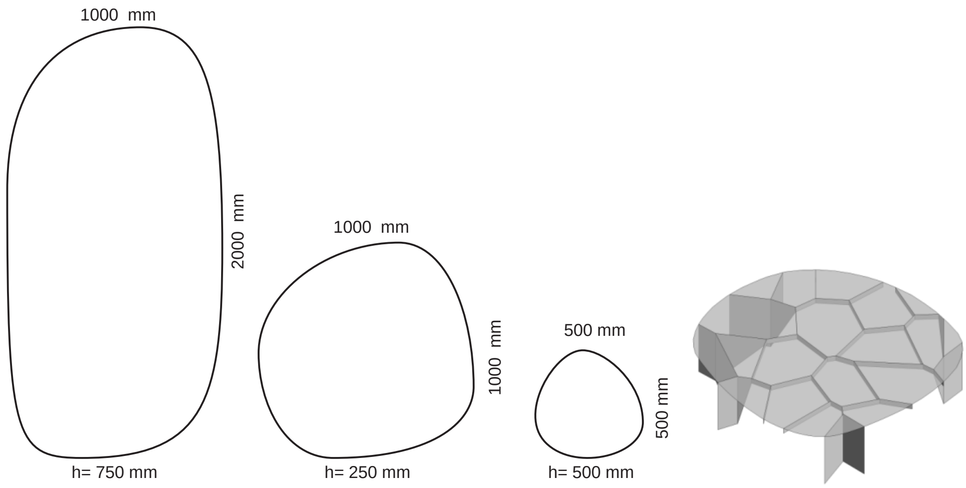
Based on the key elements, a number of members of each element category could be identified. Of course, an exhaustive search for every possible member of each element would not be realizable within a short timeframe. The search for element members was therefore concentrated to finding only a few members of each element category, but they should still be possible to combine into a large number of products. The rationale behind this was that if one can find a large number of feasible products from a very limited set of element members, it should be possible to find many more as the elements are further surveyed, thus showing that the domain of application is extensive for the approach. The study showed that one can identify at least one feasible set of element members enabling the approach. This set consists of 2.5D products such as bookshelves, room dividers and tables designed with 2D tessellations such as the Voronoi diagram or isohedral tessellations, and produced from flat sheet materials such as plywood and stainless steel by CNC production technologies such as laser cutting, water jet cutting, and bending. Even in this small set of element members there is a wide range of feasible combinations, which is a strong argument in favor of the relevance of the approach.

4.2 Paper II – An approach to constraint-based and mass-customizable product design

In Paper II, a first design tool based on the approach was developed that would: 1) allow for an improved integration of industrial design into the product development process, 2) expand the morphological repertoire of designers, and 3) result in significantly improved prospects for mass customization.

For this paper, a table generation system was developed: a system that generated tables whose support structures were based on Voronoi diagrams (see Paper I for a more in-depth explanation). The generative design system ensured that the support structure of each table fulfilled structural and manufacturing constraints while minimizing the cost of production. The user could customize parts of the object, some of which were required for the initialization of the optimization process (material, contours, morphology), and could then interact with the optimization system by selecting the resulting products according to her or his preferences. The optimization and constraint handling systems were based on a genetic algorithm. The individuals of the genetic algorithm were the support structures and their genetic representation derived from the control points for the Voronoi diagram from which they were created. With an integration of morphologies with engineering and production constraints, the iterations between designers and engineers are reduced. This approach also allows for a true mass customization without resorting to rapid prototyping or pre-manufactured modules.

Three different table top contours (Figure 4-2a, Figure 4-2b, Figure 4-2c) were used to test the application. Two runs were performed before the final tables were chosen. The first search for suitable individuals used a population of 50 individuals and 600 generations (a moderate number of generations for truss problems, as pointed out in Giger & Ermanni, 2006). The search took approximately 1.5 hours of CPU time on a single-core 3.0 GHz processor. After the first search was done, the user was presented with the solutions found and was given the choice to select one or several individuals to continue with. The selected individuals were then further optimized for another 600 generations in separate searches; the resulting best individuals from the different populations were then presented to the user for a final choice. An example of the result for the coffee table is presented in Figure 4-2d. The resulting table structures were studied in detail in ANSYS Workbench, which confirmed that the structural constraints were fulfilled. Prototypes of all three tables were built (see Figure 4-3) and were exhibited at the Form/Design Center in Malmö in 2010, the 2010 DMY design festival in Berlin, the 2011 Stockholm Furniture Fair, and the 2011 100% Design fair in London.



a) b) c) d)

Figure 4-2. a) Design of the dining table top contour; b) design of the coffee table top contour; c) design of the side table top contour; d) the final optimized structure of the coffee table

Figure 4-3. Photo of the coffee table with powder coated stainless steel and brass

4.3 Paper III – Complex product form generation in industrial design: A bookshelf based on Voronoi diagrams

In this paper, the Renaissance 2.0 approach was applied to the development of a generative design tool for the design of a bookshelf, whose structure was also based on the Voronoi diagram. A genetic algorithm similar to the one used in Paper II was used to provide a semi-interactive optimization. The general approach presented in Paper III was a development of the prior work presented in Paper II. The main issues explored in this paper were how such a tool would be received by designers, and whether the approach is applicable to another product using another manufacturing system and material. An image of the interface developed for Paper III is shown in Figure 4-4.

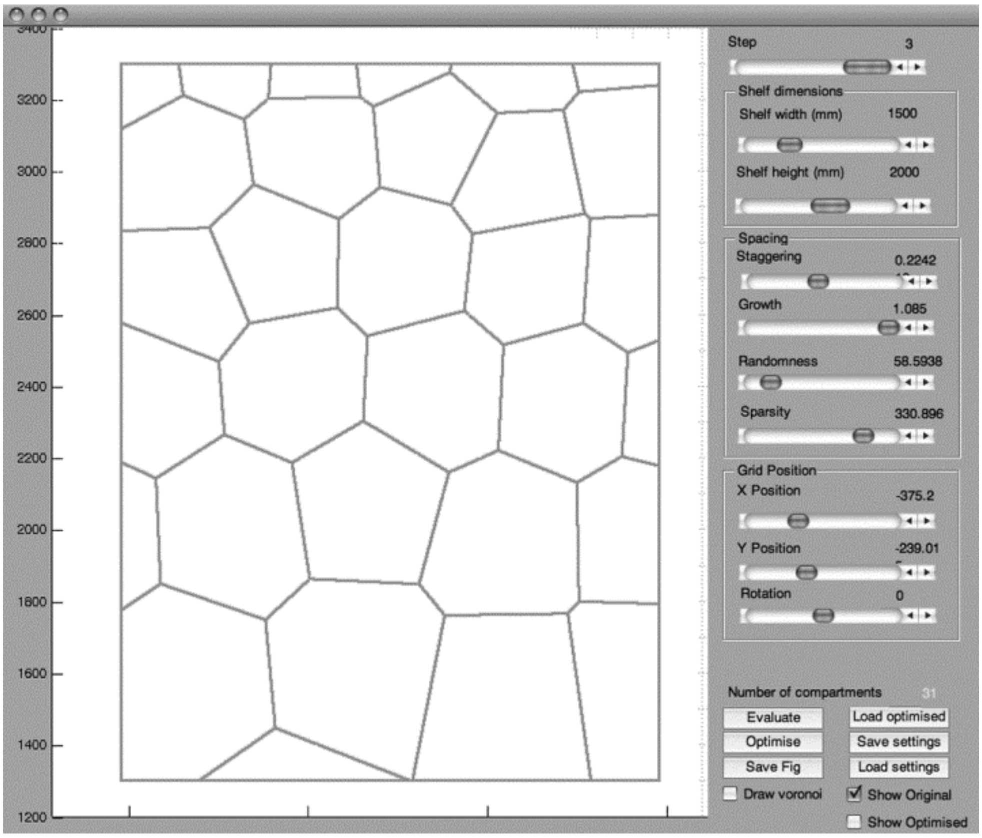


Figure 4-4. The partial control interface

Complex morphologies are difficult to handle; consequently, an explorative study on that theme was performed. It suggested that two rather opposite user attitudes prevail: one faction of peer designers preferred maximum freedom of creation, that is maximum control of the form creation process, while the other faction preferred the application to generate a bookshelf based on their functional needs (e.g. adapt it to the number and types of objects to be stored) and would ask for a "surprise me" effect for the final solution.

A study of the feasibility of optimizing economic and functional objectives while respecting engineering constraints was also performed. The user could specify the dimensions of the shelf, and the desired number of compartments in it. The shelves were subject to production constraints, and it was desirable to generate a shelf with a high percentage of compartments useful for stacking books at the lowest possible price.

Two shelves (700x2000 mm and 2000x2000 mm, 8 mm plywood) were generated using the application and are shown in Figure 4-5. They were optimized using a population of 50 individuals during 300 generations, which took around 1 hour on a dual-core 2.2 GHz processor.

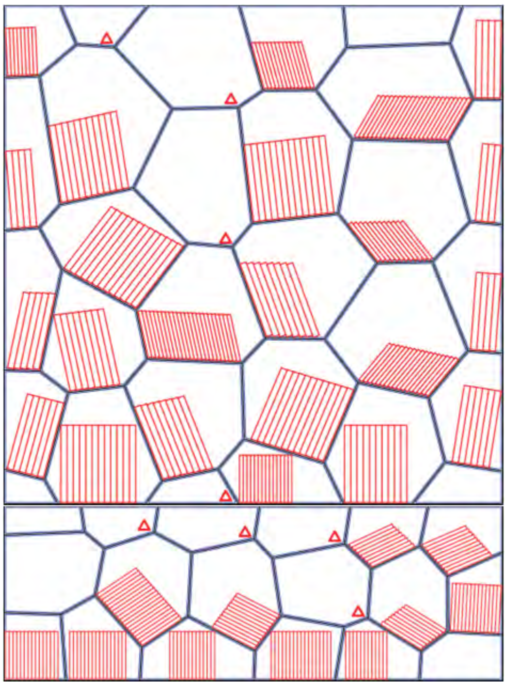


Figure 4-5. Illustration of the final bookshelves with stacked books

To verify that the bookshelves were structurally sound and stable, they were tested by a finite element analysis. The shelves were subjected to a load of 10 kg in each compartment to simulate books. The material used was 8 mm medium density fiberboard, an engineering wood product that has properties similar to plywood, but is isotropic, which makes the analysis simpler and more conservative, with the modulus of elasticity being 8 GPa, Poisson's ratio being 0.3 and density being 500 kg/m^3 (Kretschmann, 2010, pp. 5–3–5–8). The analysis was done in ANSYS Workbench. The structural analysis of the shelves indicated that the maximum deformation of either bookshelf never exceeded 0.4 mm. The results are shown in Figure 4-6 and Table 4-1.

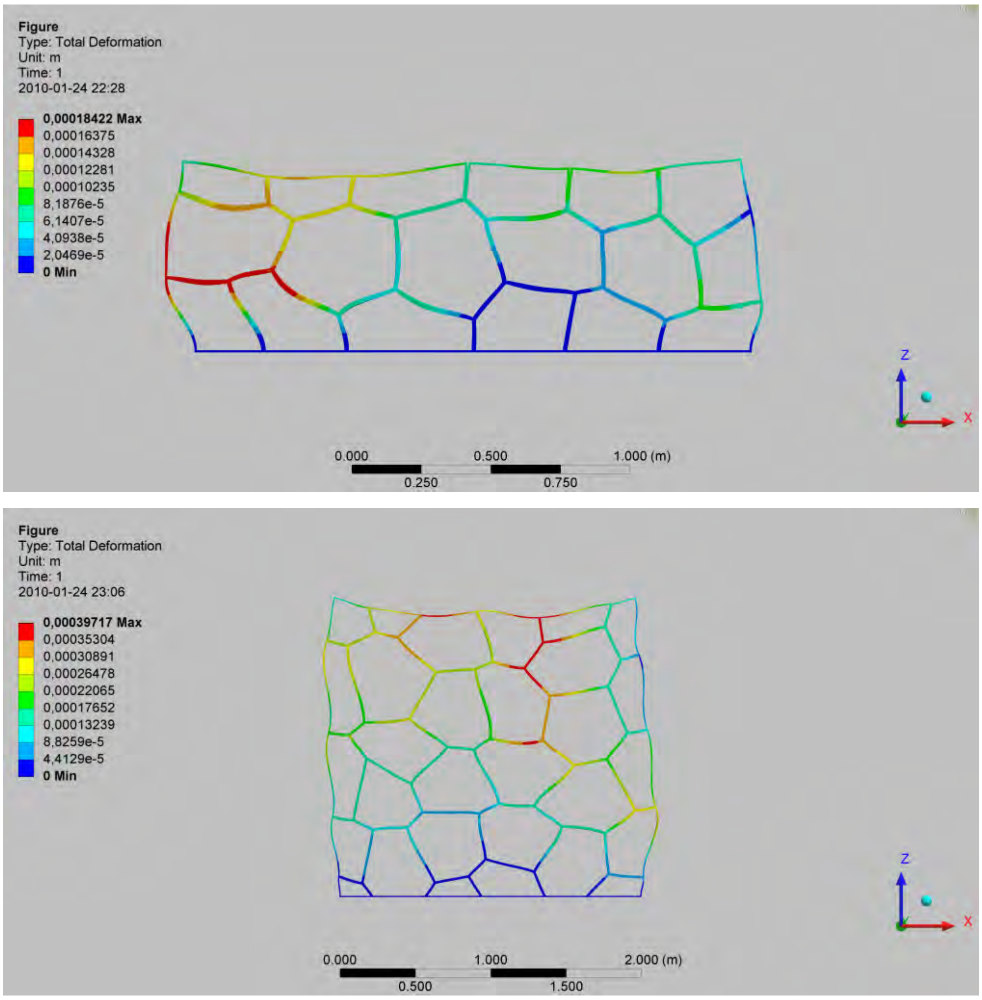


Figure 4-6. Structural analysis of the two bookshelves using ANSYS Workbench

Table 4-1. Results from the analysis in ANSYS Workbench

Bookshelf	Volume(m^3)	Weight(kg)	Max. deformation(m)
700x2000	5.38E-02	26.911	1.84E-04
2000x2000	0.12252	61.259	3.97E-04

To verify the aesthetic qualities and their commercial potential, the two bookshelves were subsequently 3D-modelled in 8 mm plywood, then photo-realistically rendered based on the output geometry of the application and subsequently shown to the peer designers. To confirm the bookshelves' producibility, drawings were produced from the output, and two prototypes were built and exhibited at the 2010 DMY design festival in Berlin (Figure 4-7), the Form Design Center in Malmö, Sweden in 2010, and the 2011 100% Design fair in London. The general impression was that they were well received by peer designers.

Figure 4-7. Photo of the two bookshelf prototypes

4.4 Paper IV – Strategies for consumer control of complex product forms in generative design systems

In order to investigate the acceptance among consumers of generative design tools for mass-customization, Paper IV describes the development of a third design tool for the design of bookshelves and tables. Building on the conclusions of Paper III, the design tool had two separate interfaces: one with no control of the form generation, and the other with total control and full feedback of the producibility and stability of the design (see Figure 4-8). The separation was made in order to be able to test if consumers would prefer an automated system to create the design for them, or if they would like to have full control over the process. The evaluation occurred at a design exhibition (at the Form/Design Center in Malmö, Sweden, see Figure 2-2) attracting visitors with a strong interest in design and furniture. The volunteers desiring to test the system could use either the total control or no control setup. 17 participants spontaneously chose the total control setup and 9 participants chose no control. A double-sided binomial test showed that there was no significant preference for one setup over the other (p = .08). With 26 participants and a power of .80 (and α = .05), we can rule out that one of the two setups is chosen in more than 76% of the time.

In order to complete the test, 7 more participants used only the no control setup (they were not made aware of the total control setup so this did not affect their evaluation). In total, 17 participants tested the total control setup and 16 the no control setup. In the first setup, the satisfaction score on a scale from 0 (not satisfied at all) to 1 (extremely satisfied) was .77 with a standard deviation (SD) of .15, in the second setup .84 (SD = .13). The double-sided t test of the difference between means did not produce a statistically significant result (p = .13), which means that the study could not statistically show that the participants appreciated one setup more than another. In the no control set-up, although the user could not influence the final result much, the perception that the product is tailored to individual needs and expectations was not altered.

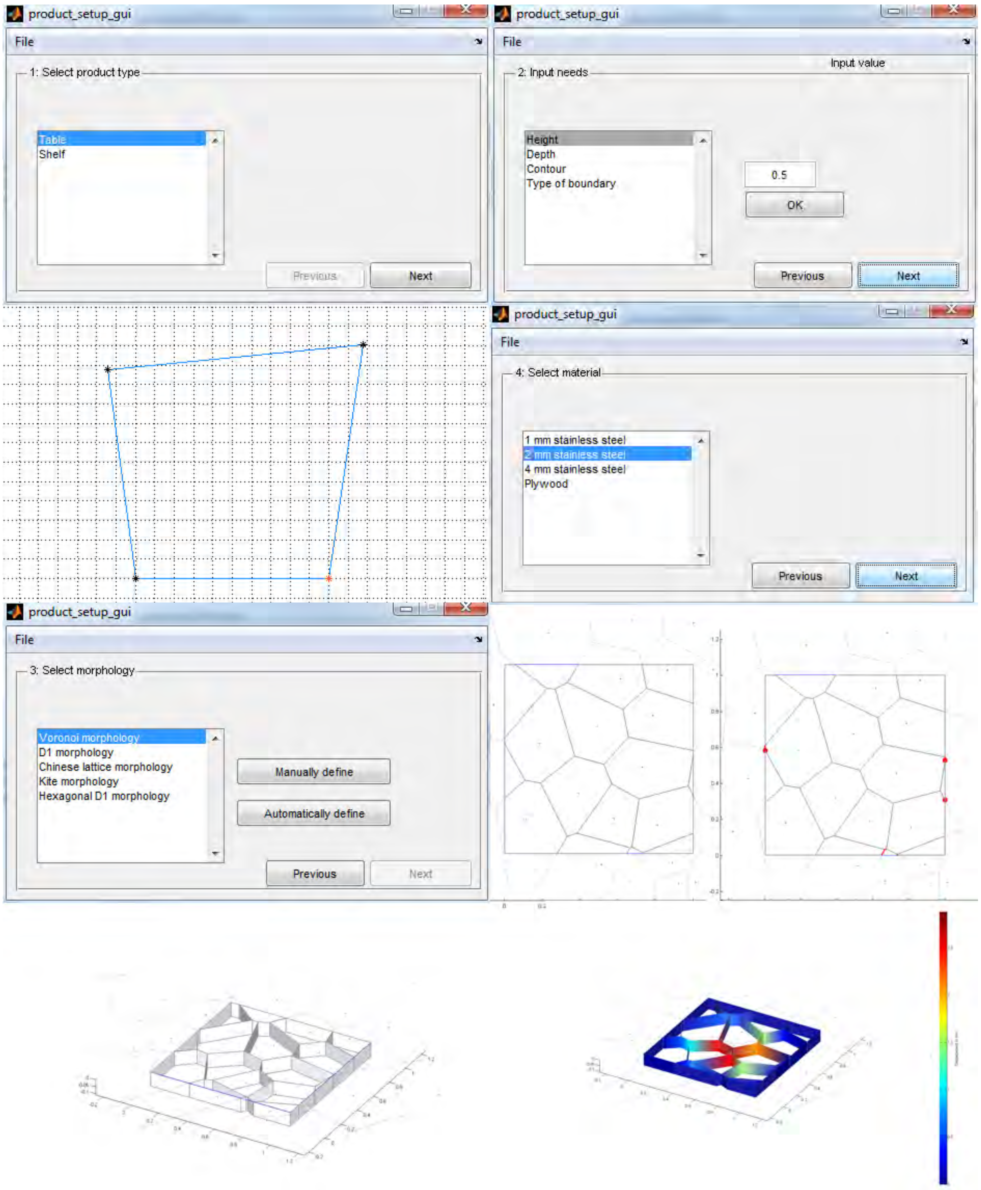


Figure 4-8. Total control-set-up

4.5 Paper V – Generative design systems for the industrial design of functional mass producible natural-mathematical forms

In order to show the feasibility of using 3D morphologies, and extend the number of verified production methods, Paper V explored two new morphologies and one new material system. It was also important to demonstrate, as noted in Section 1.5, that the output of nature-based computational means of form generation does not have to be confined to rapid prototyping, but can also be realized with established fabrication technologies allowing for mass production – whether using high or low tech materials. The forms are complex in the sense that creating them manually would be very time consuming and difficult; the constraints and objectives to which the forms must adhere further adding to the complexity of generating feasible forms.

The paper examined: 1) Lindenmayer systems (L-systems) (Lindenmayer, 1968; Prusinkiewicz, Hanan, & Lindenmayer, 1990) coupled to a genetic algorithm to create user-controlled branching support structures (see Figure 4-9 and Figure 4-10), and 2) minimal surfaces (Isenberg, 1992, pp. 1–5; Dierkes, Hildebrandt, & Sauvigny, 2010) to create user-controlled lighting diffusers (see Figure 4-11, Figure 4-12 and Figure 4-13). These morphologies were chosen based on their functional and aesthetic properties, and to represent two different adaptation processes. Both these applications have a set of constraints and objectives associated with them in terms of functionality and manufacturability. To verify this process, a set of lights based on the minimal surfaces were built and were later selected for exhibition at several international design fairs (DMY2011, Stockholm Furniture Fair 2013, Biennale Internationale Design Saint-Etienne 2013) showing that the generative design system made it possible to produce a fully developed, "ready-for-sale" product, with potential for large-scale production. This is a step towards enabling designers the same level of form articulation as has been available to artists and architects, even though the constraints on the design activity are much different. The paper demonstrated, through the implementation of two design systems, how fairly complex everyday objects based on 3D natural-mathematical morphologies can be designed, evaluated and produced using mass production techniques; that digital and analogue methods can be linked to create an aesthetic and functional whole beyond purely decorative mimicry.

Figure 4-9. Structure mock-up of one of the generated L-systems

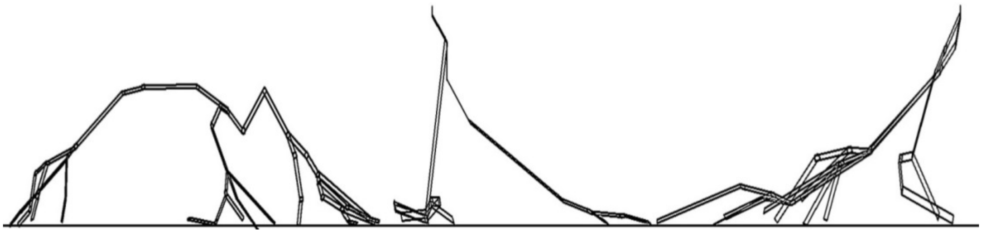
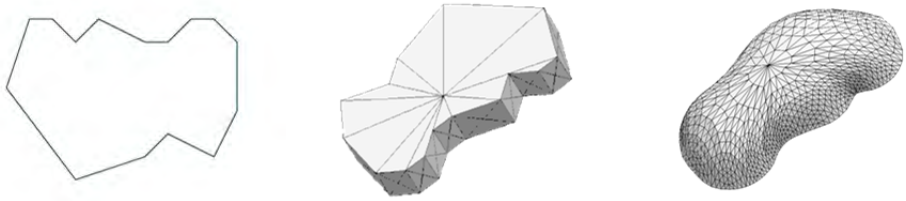
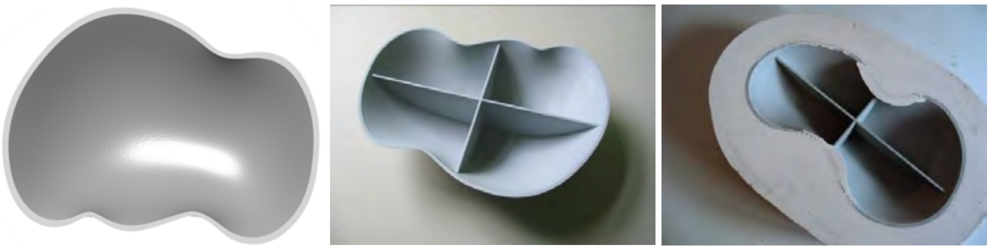


Figure 4-10. Output of the optimization process showing three different solutions that satisfy the constraints



(a) (b) (c)

Figure 4-11. a) User defined drop contour, b) initial extruded surface block, c) first surface energy minimization step



(a) (b) (c)

Figure 4-12. a) Thickened mesh, b) 3D printed model with support ribbing, c) mold-making

Figure 4-13. Final pieces after firing

4.6 Paper VI – Constraint-handling techniques for generative product design systems in the mass customization context

This paper builds on the results regarding sequential constraint-handling techniques in Paper II and (Motte, Nordin, et al., 2011), and can be seen as a validation of the key aspects of the constraint-handling approach. As noted in Section 1.4, engineering design problems are most frequently characterized by constraints that make them hard and time-consuming to solve. When evolutionary algorithms are used to solve these problems, constraints are often handled with the generic weighted sum method or with techniques specific to the problem at hand. Most commonly, all constraints are evaluated at each generation, and it is also necessary to fine-tune different parameters in order to generate good results, which requires in-depth knowledge of the algorithm. The sequential constraint-handling techniques seem to be a promising alternative, because they do not require all constraints to be evaluated at each iteration and they are easy to implement. They nevertheless require the user to determine the ordering in which those constraints shall be evaluated, which is problematic since one either needs to test all possible permutations of the constraint ordering (which, depending on the number of constraints, can be time-consuming), or develop a heuristic for finding an optimal, or at least satisfactory, ordering. For that purpose, two heuristics, originally described in (Motte, Nordin, et al., 2011), that do not require expert knowledge from the user were devised: either choose the sequence where all constraints are ordered according to their evaluation time (H1), or choose 6 or 7 sequences randomly and test them (H2). Two sequential constraint-handling techniques (Lexcoht and BM) using the heuristics were tested against the unweighted sum technique, UWS, (Motte, Nordin, et al., 2011 showed that the weights had a minimal effect on the convergence time) with an application based on a design problem. The convergence times of the three constraint-handling techniques and two heuristics were compared; their reliability (convergence rates) were measured: it is important that the algorithm converges as often as possible to ensure that the users obtain viable solutions; finally, the diversity of the proposed solutions (how much the solutions differ from each other in terms of form) was investigated as it is important to be able to offer the user a diverse set of forms to choose from.

Concerning diversity, the investigation revealed that the intra-population diversity was not high enough to be used for presenting several alternatives to the user. It was also shown that the specific mating scheme that is built-in in the BM method did not ensure enough intra-population diversity. On the contrary, the inter-population diversity was always high; see Figure 4-14 for a comparison between the intra- and inter-population diversities. Diversity is thus gained at the price of more runs, but if this result is confirmed, it will have three important positive implications: 1) whatever the constraint handling technique, diversity would be ensured; 2) one would not have to check for diversity before presenting solutions to the user; 3) one would not even need to define a diversity measure, which in itself can be a very complex problem.

The constraint-handling technique that had the best convergence time was the Lexcoht method using H2 (see Figure 4-15). Together with the results presented in (Motte, Nordin, et al., 2011), this confirms that the sequential constraint-handling techniques are promising for the kind of problem presented here. The different sequences ensuring the best convergence time need to be tested first (H2), but the gain is substantial. It is important to mention that the convergence time distributions are highly positively skewed, so a good constraint-handling technique not only allows for a quicker convergence but also avoids very lengthy runs. The parameters for the BM techniques that were set according to the recommendations from (Schoenauer & Xanthakis, 1993) yielded good results. If the sequences cannot be tested, the evaluation time-based heuristic (H1) also yields good results.

The convergence rates were significantly better for Lexcoht with some sequences; other sequences were on a par with the weighted sum technique. Convergence rates were poor for the BM method in this setup, but were excellent in (Motte, Nordin, et al., 2011) — 100%. The calculations can be made in parallel, and in multi-core or cluster setups the number of runs needed may not be critical.

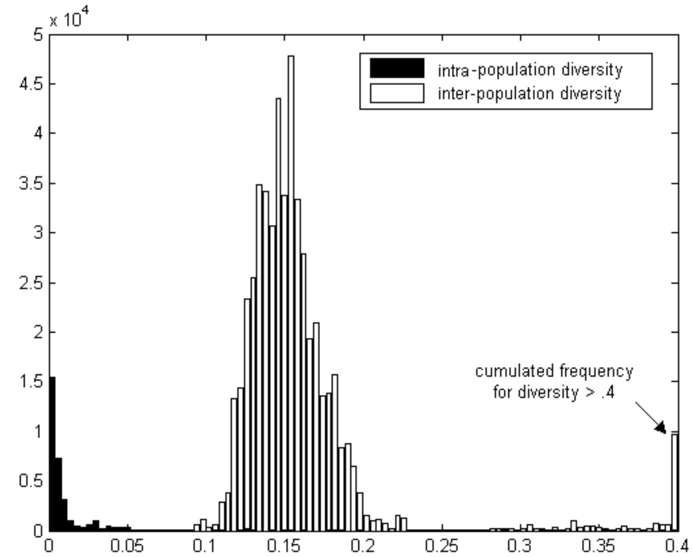


Figure 4-14. Histogram of the intra- and inter-population diversities. The inter-population diversity is almost always superior to the intra-population diversity

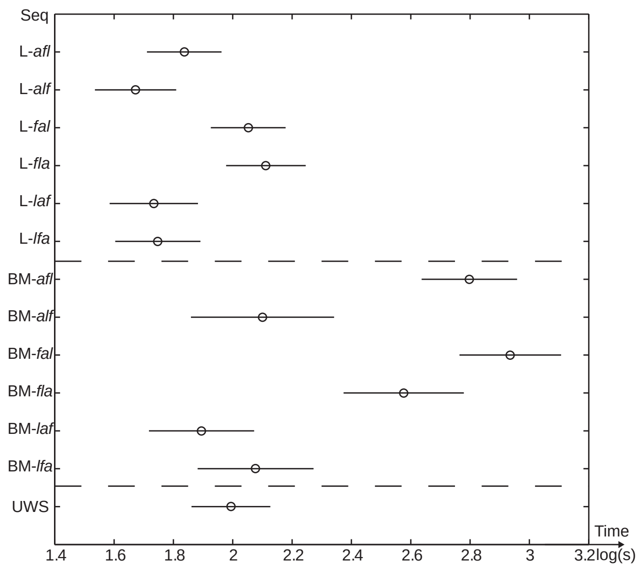


Figure 4-15. Diagram of the convergence times for the different methods

4.7 Paper VII – Restart strategies for constraint handling in generative design systems

The results from Paper VI showed that the convergence times varied drastically between runs, indicating that the constraint handling method sometimes happened upon a solution that fulfilled all constraints during the first few iterations, while at other times it was trapped in an unfavorable part of the search space. Based on the highly unpredictable convergence times, spanning several orders of magnitude (see Figure 4-16), a strategy based on restarting the search for viable solutions after a certain number of iterations was devised. However, determining when to cut off and restart the search is not a trivial task; selecting a low cutoff value decreases the probability of finding a solution within a single search, which increases the number of iterations necessary for convergence. Moreover, as the search is conducted on an engineering problem, where the probability to converge to a solution within a certain convergence time is unknown, it is not possible to find the optimal cutoff value analytically. Therefore, an algorithm for determining when to restart the search needs to be either independent of the problem, or be able to adapt the cutoff value as the search progresses. In this paper, two strategies are investigated for such selection, and their performance is evaluated on two constraint-handling techniques for a product design problem. The results show that both restart strategies can reduce the overall convergence time by over 90%. Moreover, it is shown that one of the restart strategies can be applied to a wide range of constraint-handling techniques and problems, without requiring any fine-tuning of problem-specific parameters. As diversity of the solutions is important in a generative design context, a concern was that, due to the way the restart strategies favor more easily reachable solutions, the solutions could be quite similar, even though the results in Paper VI indicate a high diversity between solutions from separate runs (interpopulation diversity). In order to investigate how restarting affects the diversity of the solutions in this application, the diversity measure used in Paper VI was applied to the solutions. The evaluation of the diversity of the solutions showed that the restart strategies were not prone to finding the same solution repeatedly (see Figure 4-17). The diversity was comparable to that of the baseline constraint handling techniques.

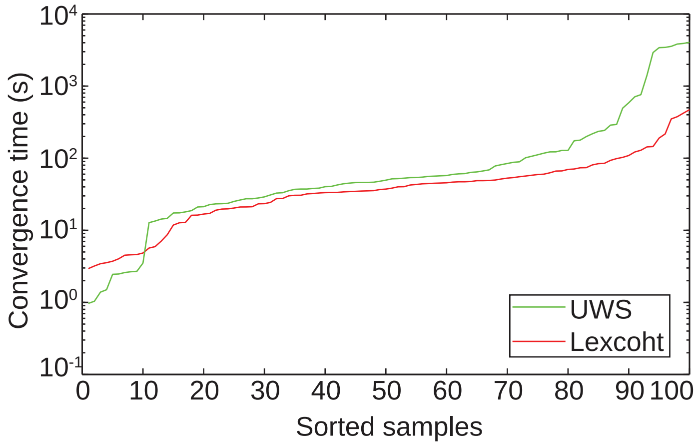


Figure 4-16. Sorted convergence times for the two constraint-handling methods (log. scale)

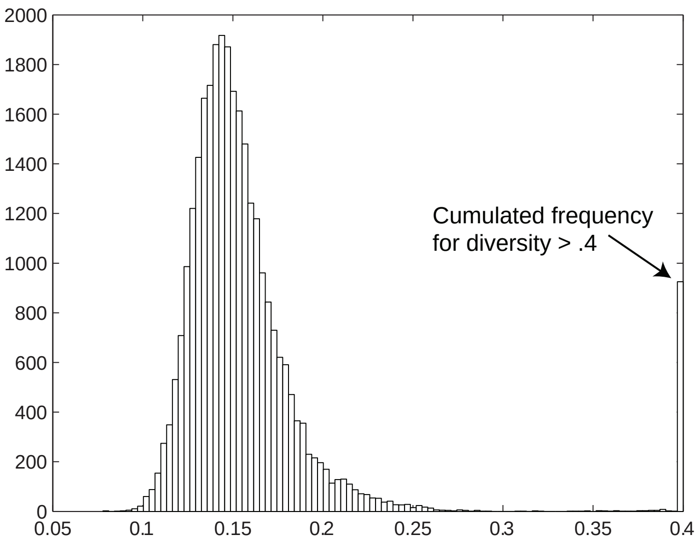


Figure 4-17. Histogram of the diversities of 250 solutions

4.8 Paper VIII – Challenges in the Industrial Implementation of Generative Design Systems – an Exploratory Study

The results from the previous papers show that the approach is technically feasible, that is there is a large domain of applications for the approach, and constraints, objectives and user interaction can be successfully handled. However, in order for the approach to be useful not only academically, it needs to be feasible to implement industrially. In Paper VIII, two case studies conducted while developing generative design systems for companies were described. The aim was to document the challenges associated with the development of generative design systems in practice, thereby aiding the elaboration of recommendations for future development. The first case study focused on an engineering design application in a company providing solutions for dispensing metal discs (see Figure 4-18). The implementation of the design tool entailed developing an evaluation function capable of simulating thousands of rigid bodies in motion to be able to measure the performance metrics, which were also established together with the company during the project. The second case study focused on an industrial design application in a design consultancy company working on, amongst other things, projects related to the design of surveillance cameras (see Figure 4-19). The development project consisted in creating a custom thermal evaluation code and several shape generation algorithms together with the industrial designers.

The results showed that there were a number of challenges. Overall, the challenges identified are not related to whether the design problems are artistic or technical in nature, but rather to the systematization of parts of the design process. For instance, it can be difficult for a company that has never dealt with generative design tools before to fully grasp the possibilities and utilize them in an optimal manner. Although there are many benefits with a design tool tailored to the design process, it requires a substantial investment of resources that could potentially be better spent elsewhere. It is therefore of importance that the industrial application is selected with the unique capabilities of generative design systems in mind.

Another identified challenge was how to deal with the designer not being familiar with programming. In the ideal case, the designer using a generative design system is also the programmer. However, even in that scenario, designing an algorithm instead of an object is a challenging task, or as Knuth puts it regarding the design of a shape generation algorithm for fonts, *"Meta-design is much more difficult than design; it's easier to draw something than to explain how to draw it"* (Knuth, 1995, p. 1).

A challenge, which was also noted in the conclusions of Paper II, was the difficulty of knowing what should be included in the design tool. Developing an algorithm for doing design tasks that are almost solely based on the designer's subjective opinion can be time-consuming as the designer is refining his or her vision of the product based on the output of the algorithm. Moreover, a more specialized tool will not be as applicable to other products, which could have made the initial investment more economically justified.

Finally, challenges relating to the formulation of objective and constraints for the design generation were also identified. In order for the optimization or automatic generation of a product to be possible, the objectives and constraints associated with the product must be possible to measure, either through virtual or physical tests, or through user feed-back. The problem of finding suitable metrics is not unique to generative systems, the general recommendation in for instance Ulrich and Eppinger (2012) and Ullman (1997) is for the product specification to be based on measurable metrics and target values. However, in practice, this might not always be strictly followed since the company could believe that the investment in determining metrics and developing methods for evaluating them is not worthwhile compared to simply basing the evaluation on trial and error or the experience and intuition of the designers.

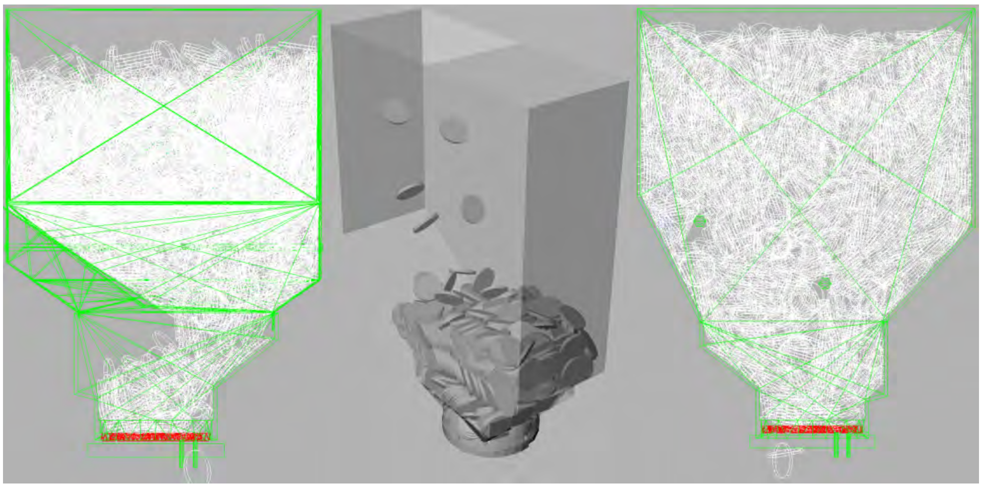


Figure 4-18. Design concepts of metal disc dispensers

Figure 4-19. Design concepts for surveillance cameras

5 Conclusion and future research

5.1 Conclusion

The question whether form should follow function, rather than being used purely for aesthetic reasons, often leads to a rather polarized discussion, even though in practice, designers are rarely quite so dogmatic in their work (Crilly, Moultrie, & Clarkson, 2009, p. 225). It does, however, capture a real design issue in that either form or function often takes precedence in a design. With the approach developed in this thesis, the aim has been to reconcile the two extremes by allowing both form and function to be developed simultaneously without one being more important than the other. The approach presented in this thesis contributes to: 1) increasing the integration of industrial design into the product development process, 2) expanding the morphological repertoire of the designers, and 3) enhancing the possibilities for product form customization by customers. The issues relevant to the industry have been tackled separately in the literature, but have not been integrated. The proposed approach is to develop generative design tools that allow a user to design products with complex forms, while assisting them in ensuring that the user and engineering constraints and objectives are met. By using complex morphologies, designers can deal with forms they could scarcely imagine. Products containing these structures can then be optimized taking into account aesthetic, functional, engineering and manufacturing constraints and objectives. The user can control parts of, or the entirety of the design, such as what material to use, the contour of the object, and which morphology it will be based on, and then interact with the form manually or through an interactive optimization system by selecting the resulting products according to her or his preferences. This approach also allows for a true mass customization without resorting to rapid prototyping. The answers to the specific research questions dealing with the technical feasibility of the approach are summarized below.

What is the domain of application of the approach?

The studies in Paper I have shown that there are a number of viable technologies for implementing the approach. Judging by the wealth of products available for 2.5D product typologies coupled to 2D tessellations, CNC cutting and bending, and sheet materials, it is not difficult to imagine that many more products are suitable for the approach. For example, just in the domain of 2.5D objects they can be designed with regular or irregular tessellations: furniture, flooring and wall elements are obvious examples; other are façade elements (window grates, balustrades), enclosure elements (wind deflectors, noise barriers), driveway elements (drainage gates, banisters), etc. The feasibility of the combination of the key elements found in Paper I was subsequently tested in Papers II, III and IV through the implementation of three design tools, production of five prototypes, and the acceptance to international design fairs. The feasibility of 3D morphologies and more complex products was shown in Papers V and VIII.

Based on these results, there does not seem to be any technical limitation in the type of product typology, morphology or material system that can be used with the approach. In practice, the limitation is rather based on the economic feasibility of implementing a generative design system. A design system should ideally be implemented in applications requiring many variants of the same product to be created, in case customization of the product is important, or if the same generative system can be reused for many other, similar, products in the future. It could lead to new business strategies and models, building on augmented design automation and customer involvement instead of the traditional business model development-manufacturing-distribution-consumption.

How can a product form be optimized with respect to both qualitative and quantitative aspects?

The major task for a product designer is to create a product that the customer will find interesting in terms of function, aesthetics, and price, while respecting engineering constraints. Papers II and III present a method based on genetic algorithms for optimizing the form of tables and shelves to maximize or minimize certain economic and functional objectives, such as the cost of the tables and the cost and usefulness of the compartments of the bookshelves. In Paper V, the approach is further tested by applying it to new products, and 3D morphologies, while in Paper VIII, the challenges of industrial implementation are investigated. The results show that optimization of both user-related and technical objectives can be done successfully, even for complex multi-objective problems. In practice, the most difficult part is to find and formalize experience-based rules into objectives and constraints to ensure that the optimized result matches the vision of the designer. The problem of developing a list of product specifications is not unique to generative design systems, but the rigor with which they need to be formulated to avoid designs that a human designer would easily spot as unfeasible necessitates substantial testing. Moreover, certain specifications might require specialized evaluation functions to be developed if there are no currently available solutions.

How can the engineering constraints be handled in such a way that diverse solutions can quickly and reliably be proposed to the user using a generic method?

The constraint satisfaction problem contains a mixed (continuous and discrete) set of constraints; therefore, stochastic search algorithms like genetic algorithms are recommended for finding solutions.

The solution space for engineering problems is often small and sparse, and some constraints are either hard or time-consuming to fulfill (e.g., require extensive finite element analyses). The sequential generic constraint-handling techniques investigated in (Motte, Nordin, et al., 2011) and Paper VI are promising to the extent that they take into account the occasionally lengthy constraint evaluation times. If the generative design tool is to be used repeatedly, an initial experiment with random ordering of the constraints can be used to find a constraint sequence that is close to optimal. If the tool is only to be used once, or, if there are large differences in the evaluation times of the constraints, the lexicographic sorting should be based on the evaluation time of the constraints. In either case, the results in Paper VII show that an adaptive restarting strategy should be used to ensure that unnecessary time is not wasted on unfeasible designs or trying to escape from local optima.

The generative design system must also be able to present a large diversity of product alternatives for the user to choose from; this is efficiently managed by launching several runs in parallel rather than using diversity-promoting techniques in the population selection, as shown in Papers VI and VII.

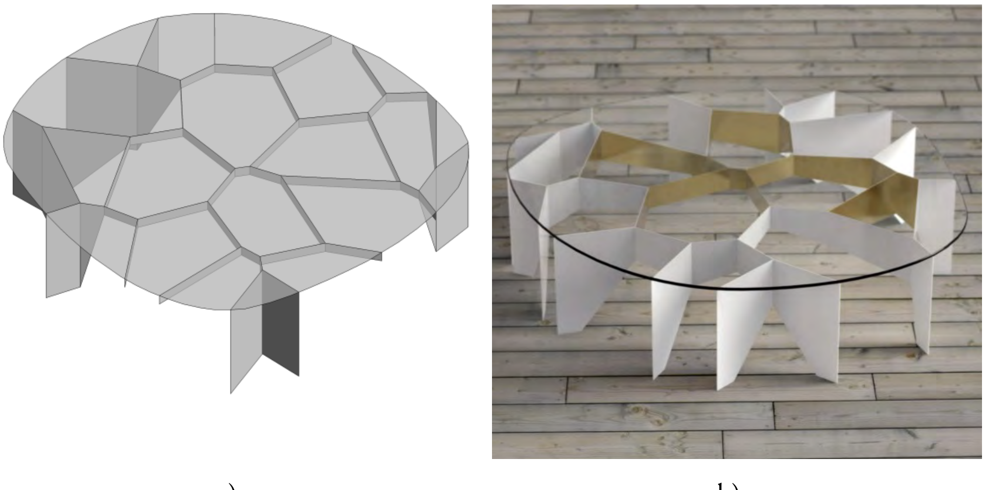
What are the possible modes of interaction with the generative design tool for the user?

The manipulation of complex morphologies is not straightforward, but the users appear to enjoy both the total control and no control setups, as presented in Paper IV. It seems that a solution with two different setups offers the user the most satisfaction, as opposed to a solution with only one in-between mode, as presented in Paper III. However, the details of each setup, especially the total control setup, need to be carefully worked out. In the bookshelf example, should users be able to specify horizontal walls (if they want to put a vase in one cell) or put some virtual books on the shelves they designed, etc.? The same issue was observed in the case studies in Paper VIII, where a considerable portion of the development time for the industrial design application was spent on finding a satisfactory shape generation algorithm together with the designer.

Moreover, it may be necessary to differentiate in terms of amount and type of control between the online systems (for customers), offline systems (for designers) and consulting systems (customers and designer co-designing in a shop or Web shop).

The development of a product design system is time-consuming and complex in itself; even if the system can be made relatively generic, some modifications will always be necessary. This requires that the morphologies, material and production systems be chosen in advance. As is always the case with the development of a system, iterations will be necessary. It is also not very beneficial to devise such a system (to avoid the iterations design-engineering-production) if one does not plan to re-use it several times. This would only shift the discussions from the design of the product to the design of the system.

A case in point is that of the development of the table with the Voronoi diagram. The system was first developed with the walls of the Voronoi cells of equal heights (Figure 5-1a). However, it was decided after seeing the final result that the design would be more interesting aesthetically with varying cell heights (Figure 5-1b). This means that the optimization algorithm needs to be changed accordingly in order to optimize for weight and stiffness. This is an inherent problem to the approach, and it must be dealt with in any system development. The same observation was made in Paper VIII, where it was difficult to determine what should be part of the design tool, and what parts should rather be performed manually. The balance between automation and development time needs to be carefully maintained during the implementation of an industrial tool.



a) b)

Figure 5-1. a) The final optimized structure of the coffee table, original design (constant cell wall heights, apart from the table feet). b) Rendering of the final design (varying cell wall heights)

What challenges are present in developing a generative design tool in an industrial project?

The question was investigated in Paper VIII through two case studies. In order to fully answer research question five, a much larger study would have to be performed to include all organizational aspects and product development processes (see Section 5.2 and Figure 5-2). However, the study does offer an in-depth view of a number of hurdles that most likely will be encountered in similar projects. The results showed that most of the challenges identified could be placed into one of two categories, which are described below.

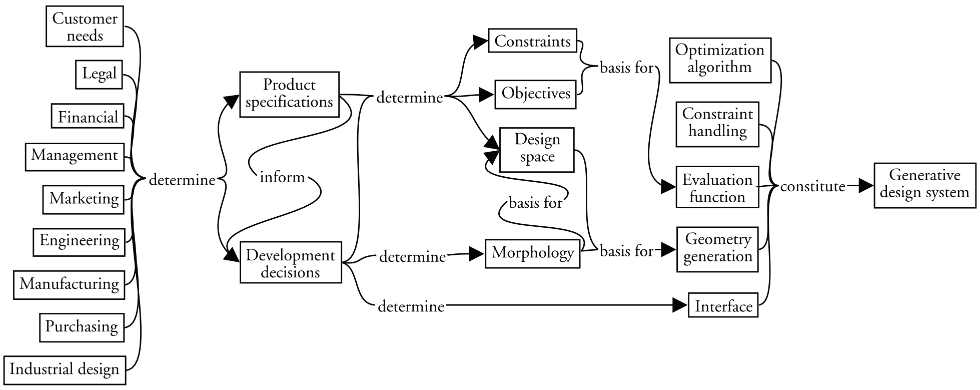


Figure 5-2. A schematic of how a generative design system fits in with the overall product development process

What challenges are present in determining the constraints and objectives of a generative design system?

The results from the industrial case studies in Paper VIII show that a major part of the development time is required for determining appropriate constraint and objective formulations, even though both companies had successfully developed similar products previously, indicating that they had an in-depth understanding of the products. Generally, the rules first needed to be found out through interviews. Based on the aspect of the product's performance the design rule was put in place to improve, a metric possible to measure through simulation needed to be formulated. The metrics then needed to be implemented in the evaluation function, and fine-tuned.

Additionally, development time will most likely have to be spent on straightening out fringe cases and bugs in the evaluation code before a successful optimization or concept generation can be completed. This is of course not something that is unique to generative design systems, but rather something inherent in the product development process and any global optimization problem with a large design space. However, one should be aware of the extra time that will be required to fully specify a product if it is to be design using an automated tool.

How does the process of creating and modifying a design tool work in practice when working with a designer in an industrial project?

In Paper VIII, the case study on the consumer electronic product showed that even though the concept of generative design was understood quite quickly, actually using it and understanding the possibilities provided required a tool to be developed as a demonstration of its capabilities. The strategy adopted in the case of the design consultancy company, to push them towards applications more in line with the capabilities of generative design systems, was to discuss what possibilities could arise if the technical parts should become an integral part of the product expression. This is perhaps something they have tended to avoid because of the extra complexity, but by freely generating ideas, without considering the feasibility of implementing them, a number of new applications could be found, amongst them the application described in this paper.

Another challenge that is likely to occur in similar projects is that the designer is unable to input new shape generation logic directly. Letting the designer describe the shape generation logic to the programmer, in a way, moves the bottleneck from the designer interacting with the engineers who evaluate the technical feasibility of their concepts, to the designer interacting with a programmer, and lacks some of the benefits that generative design has to offer. However, a large part of the designer's task is to continuously evaluate the form based on a set of criteria and being able to justify their design decisions, which is conceptually not very different from defining the logic an algorithm should follow to create the form of a product.

5.2 Future research

This thesis has explored the domain of application and the technical feasibility of the Renaissance 2.0 approach, but to ensure a commercially viable framework much work is still to be done. Future studies should include:

Further development of the generative design system

In computationally intensive simulations, it is quite common to have some constraints evaluated simultaneously. For example, constraints such as maximum stresses, displacements or buckling can in some cases all be evaluated using a single finite element analysis. For such linked constraints, a sequential constraint-handling technique cannot be used in the manner described in (Motte, Nordin, et al., 2011); instead, one possibility could be to treat these linked constraints as a group of constraints evaluated simultaneously using techniques such as the weighted sum; further research is needed in this area.

With sequential constraint-handling techniques, the constraint satisfaction and optimization parts of a structural problem are considered separately. Only the constraint satisfaction part has been investigated here. It is possible to transform constraints into objectives through penalty functions and use multi-objective optimization techniques, see, for example, Coello Coello et al. (2007, pp. 113–114) instead of sequential constraint-handling techniques. The relative benefits of the multi-objective optimization techniques and the sequential constraint-handling techniques require further analysis.

Further studies should investigate the use of other alternatives, such as multi-objective genetic algorithms or simulated annealing and combined methods such as global optimization techniques coupled with local optimization techniques. Using the design of experiment methods and response surface models together with design selection based on Pareto optimality are also promising areas to look into to speed up the optimization and constraint-handling process and limit the number of alternatives proposed to the user. Other approaches to reduce the time the user spends interacting with the system, such as using machine learning or other adaptive techniques to capture the design intent and subjective reasoning of the user are also important to consider; especially since they could affect the acceptance of the approach by removing some of the activities that traditionally have relied heavily on human creativity.

Further investigation of implementation challenges

The list of implementation challenges presented in Paper VIII is by no means exhaustive, but rather scratches the surface of the implementation issues that might be faced when developing generative design systems. Issues relating to the integration of generative design systems into the company's organization, data management system and development routines need to be studied before a fully mature system can be achieved, especially if the design system is to be used by a larger company with already established development processes and tools. Questions such as how traceability, verification, validation and product lifecycle management can be upheld when introducing a semi-autonomous tool also arise, especially if customization is to be part of the business model. Additionally, aspects relating to any commercial software need to be considered, such as how maintenance, licensing and reliability should be handled.

Acceptance of the approach

Because of research initiatives such as the Renaissance 2.0 program, and societal transformations such as consumer empowerment, it is quite possible that the definition of what a designer is and does will change considerably in the future; designers may eventually become scriptwriters, moderators or curators, rather than authors of their own work. This aspect has been touched upon in Paper VI, VIII and through design exhibitions. The question of how such tools will be received by designers and customers also needs investigation, as the changed role of the designer and the added customer control might not always be welcomed. Will the designer work with a programmer to create the tools they will use later to develop new concepts? Will the designer work as an evaluation function for the optimization algorithm? Will the designer just design the interface for form creation and let the customer do the actual configuring? Are the customers interested in spending time tailoring their products or do they simply want off-the-shelf solutions? To what extent do they want to be able to control the product form, and are they interested in or capable of understanding feedback about engineering constraints and objectives?

Development of adequate business models

There could potentially be many viable business models utilizing the Renaissance 2.0 approach, for instance letting only in-house designers use the tool to design classical mass-produced products, or on the other extreme letting customers themselves design the products, either for their own use or for purchase by other consumers. A few possibilities are outlined in Paper I. To enable a user to customize a product it is necessary to develop new business and product development models. No longer will there be a possibility to evaluate, prototype and test the specific design being produced. The production system will have to be made much more flexible, and so will the logistics.

References

Agarwal, M., & Cagan, J. (1998). A blend of different tastes: The language of coffee makers. *Environment and Planning B*, *25*(2), 205–227.

Agarwal, M., Cagan, J., & Constantine, C. G. (1999). Influencing generative design through continuous evaluation: Associating costs with the coffeemaker shape grammar. *Artificial Intelligence for Engineering Design, Analysis and Manufacturing - AI EDAM*, *13*(4), 253–275.

Anderson, C. (2006). *The Long Tail: Why the Future of Business is Selling Less of More*. New York, NY: Hyperion.

Ang, M. C., Chau, H. H., McKay, A., & de Pennington, A. (2006). Combining evolutionary algorithms and shape grammars to generate branded product design. In J. S. Gero (Ed.), *2nd Design Computing and Cognition Conference DCC'06* (pp. 521–539). Dordrecht: Springer.

Barricelli, N. A. (1954). Numerical examples of processes of evolution (In Italian. Original Title: Esempi numerici di processi di evoluzione). *Methodos*, *6*(21-22), 45–68.

Bentley, P. J., & Corne, D. W. (2002). An introduction to creative evolutionary systems. In P. J. Bentley & D. W. Corne (Eds.), *Creative Evolutionary Systems* (pp. 1–75). San Francisco, CA: Morgan Kaufmann.

Blessing, L. T. M., & Chakrabarti, A. (2009). *DRM, a Design Research Methodology*. London: Springer.

Bruce, M., & Bessant, J. (2002). *Design in Business: Strategic Innovation through Design*. Harlow: Financial Times Prentice Hall.

Cagan, J. (2001). Engineering shape grammars: Where have we been and where are we going? In E. K. Antonsson & J. Cagan (Eds.), *Formal Engineering Design Synthesis* (pp. 65–92). Cambridge, MA: Cambridge University Press.

Chau, H. H., Chen, X., McKay, A., & de Pennington, A. (2004). Evaluation of a 3D shape grammar implementation. In J. S. Gero (Ed.), *1st Design Computing and Cognition Conference - DCC'04* (pp. 357–376). Dordrecht: Kluwer.

Cluzel, F., & Yannou, B. (2009). Efficiency assessment of an evolutive design system of car contours. In *17th International Conference on Engineering Design - ICED'09* (pp. 189–200). Stanford, CA: Design Society.

Coello Coello, C. A. (n.d.). List of References on Constraint-Handling Techniques used with Evolutionary Algorithms. Retrieved August 4, 2015, from http://www.cs.cinvestav.mx/~constraint/

Coello Coello, C. A. (2000). An updated survey of GA-based multiobjective optimization techniques. *ACM Computing Surveys*, *32*(2), 109–143.

Coello Coello, C. A. (2002). 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*(11-12), 1245–1287.

Coello Coello, C. A., Lamont, G. B., & Van Veldhuizen, D. A. (2007). *Evolutionary Algorithms for Solving Multi-Objective Problems* (2nd edition). New York, NY: Springer.

Crilly, N., Moultrie, J., & Clarkson, P. J. (2004). Seeing things: consumer response to the visual domain in product design. *Design Studies*, *25*(6), 547–577.

Crilly, N., Moultrie, J., & Clarkson, P. J. (2009). Shaping things: intended consumer response and the other determinants of product form. *Design Studies*, *30*(3), 224–254.

Dantzig, G. B. (1951). Maximization of a linear function of variables subject to linear inequalities. In T. C. Koopmans (Ed.), *Activity analysis of production and allocation* (pp. 339–347). New York, NY & London: Wiley & Chapman-Hall.

Davis, S. M. (1987). *Future Perfect*. Reading, MA.: Addison-Wesley Publishing.

Dierkes, U., Hildebrandt, S., & Sauvigny, F. (2010). *Minimal Surfaces*. Berlin, Heidelberg: Springer.

EZCT Architecture & Design Research, Hamda, H., & Schoenauer, M. (2004). *Studies on Optimization: Computational Chair Design using Genetic Algorithms*. Paris: EZCT Architecture & Design Research.

Farin, G. E. (1993). *Curves and Surfaces for Computer-Aided Geometric Design: A Practical Code* (3rd edition). San Diego, CA: Academic Press.

Fesser, K. (2010, September). Selective evolution [in German and English]. *Form - the Making of Design*, 13.

Franke, N., & Piller, F. (2004). Value creation by toolkits for user innovation and design: The case of the watch market. *The Journal of Product Innovation Management*, *21*(6), 401–415.

Fraser, A., & Burnell, D. (1970). *Computer Models in Genetics*. New York, NY: McGraw-Hill.

Frazer, J. (2002). Creative design and the generative evolutionary paradigm. In P. J. Bentley & D. W. Corne (Eds.), *Creative Evolutionary Systems* (pp. 253–274). San Francisco, CA: Morgan Kaufmann.

Friebe, H., & Ramge, T. (2008). *The DIY Brand - The Rebellion of the Masses against Mass Production (In German. Original title: Marke Eigenbau: Der Aufstand der Massen gegen die Massenproduktion)*. Frankfurt: Campus Verlag.

Giger, M., & Ermanni, P. (2006). Evolutionary truss topology optimization using a graph-based parameterization concept. *Structural and Multidisciplinary Optimization*, *32*(4), 313–326.

Haug, E. J., & Arora, J. S. (1979). *Applied Optimal Design - Mechanical and Structural Systems*. New York, NY: Wiley.

Holland, J. H. (1975). *Adaptation in Natural and Artificial Systems: An Introductory Analysis with Applications to Biology, Control and Artificial Intelligence*. Ann Arbor, MI: University of Michigan Press.

Hopf, A. (2009). Renaissance 2.0 - Expanding the morphologic repertoire in design. In *24th Cumulus Conference 24/09* (pp. 78–85). Aalto: Cumulus.

Isenberg, C. (1992). *The Science of Soap Films and Soap Bubbles* (2nd edition). New York, NY: Dover.

Kantorovich, L. V. (1940). On an effective method of solving certain classes of extremal problems. *Dokl.Akad.Nauk.USSR*, *28*, 212–215.

Kaplan, A. M., & Haenlein, M. (2006). Toward a parsimonious definition of traditional and electronic mass customization. *Journal of Product Innovation Management*, *23*(2), 168–182.

Kelly, J. C. (2008). *Interactive Genetic Algorithms for Shape Preference Assessment in Engineering Design* (PhD Thesis). Available from ProQuest Dissertation and Theses database. (UMI No. 3328863).

Kelly, J. C., Wakefield, G. H., & Papalambros, P. Y. (2011). Evidence for using interactive genetic algorithms in shape preference assessment. *International Journal of Product Development*, *13*(2), 168–184.

Knight, T. W. (1980). The generation of Hepplewhite-style chair-back designs. *Environment and Planning B*, *7*(2), 227–238.

Knuth, D. E. (1995). *The METAFONT book* (2nd edition). Reading, MA: Addison–Wesley.

Koza, J. R., Keane, M. A., Streeter, M. J., Adams, T. P., & Jones, L. W. (2004). Invention and creativity in automated design by means of genetic programming. *Artificial Intelligence for Engineering Design, Analysis and Manufacturing - AI EDAM*, *18*(3), 245–269.

Kraft Foods. (2006). Innovate with Kraft. Retrieved August 4, 2015, from http://brands.kraftfoods.com/innovatewithkraft/default.aspx

Kram, R., & Weisshaar, C. (2003). Breeding Tables. Retrieved August 4, 2015, from http://www.kramweisshaar.com/projects/breeding-tables.html

Kretschmann, D. E. (2010). Mechanical properties of wood. In Forest Products Laboratory (Ed.), *Wood Handbook - Wood as an Engineering Material* (2nd edition, p. 457). Madison, WI: Forest Products Laboratory.

Lakhani, K. R., & Kanji, Z. (2008). *Threadless: The Business of Community*. Harvard Business School Multimedia/Video Case 608–707.

Lee, H. C., & Tang, M. X. (2006). Generating stylistically consistent product form designs using interactive evolutionary parametric shape grammars. In *7th International Conference on Computer-Aided Industrial Design and Conceptual Design - CAIDCD'06* (pp. 763–769). Piscataway, NJ: IEEE.

Lee, H. C., & Tang, M. X. (2009). Evolving product form designs using parametric shape grammars integrated with genetic programming. *Artificial Intelligence for Engineering Design, Analysis and Manufacturing - AI EDAM*, *23*(2), 131–158.

Lindenmayer, A. (1968). Mathematical models for cellular interactions in development, parts I and II. *Journal of Theoretical Biology*, *18*, 280–315.

Liu, Y. (2003). Engineering aesthetics and aesthetic ergonomics: Theoretical foundations and a dual-process research methodology. *Ergonomics*, *46*(13-14), 1273–1292.

McCormack, J. P., Cagan, J., & Vogel, C. M. (2004). Speaking the Buick language: Capturing, understanding, and exploring brand identity with shape grammars. *Design Studies*, *25*(1), 1–29.

Media, N. B. (2007). New NIKEiD studio opens at NikeTown London giving consumers a key to unlock the world of design. Retrieved August 4, 2015, from http://investors.nike.com/investors/news-events-and-reports/investor-news/investor-news-details/2007/New-NikeiD-Studio-Opens-at-NikeTown-London-Giving-Consumers-a-Key-to-Unlock-The-World-Of-Design/default.aspx

Mezura-Montes, E. (2004). *Alternative Techniques to Handle Constraints in Evolutionary Optimization* (PhD Thesis). Computer Science Section, Electrical Engineering Department, CINVESTAV-IPN, Mexico.

Michalewicz, Z., & Schoenauer, M. (1996). Evolutionary algorithms for constrained parameter optimization problems. *Evolutionary Computation*, *4*(1), 1–33.

Morel, P., Hamda, H., & Schoenauer, M. (2005). Computational chair design using genetic algorithms. *Concept*, *71*(3), 95–99.

Motte, D., Bjärnemo, R., & Yannou, B. (2011). On the interaction between the engineering design and the development process models - Part I: Elaborations on the generally accepted process models. In A. Chakrabarti (Ed.), *3rd International Conference on Research into Design - ICoRD'11* (pp. 87–95). Bangalore: Research Publishing Service.

Motte, D., Nordin, A., & Bjärnemo, R. (2011). Study of the sequential constraint-handling technique for evolutionary optimization with application to structural problems. In *37th Design Automation Conference - DETC/DAC'11* (Vol. 5, pp. 521–531). Washington, D.C.: ASME.

Mugge, R. (2007). *Product Attachment* (Ph.D. Thesis). Department of Product Innovation and Management, Faculty of Industrial Design Engineering, Delft University of Technology, Delft.

Nordin, A. (2009). *Genetic Algorithms and their Applications within Design (In Swedish. Original title: Genetiska algoritmer och deras tillämpningar inom konstruktion och design)* (Master's Thesis). Department of Design Sciences, Division of Machine Design, Faculty of Engineering LTH, Lund University, Lund.

Nordin, B., & TT. (2010). Create your furniture using the computer (In Swedish. Original title: Skapa dina möbler vid datorn). Retrieved August 4, 2015, from http://www.sydsvenskan.se/bostad/article1300439/Skapa-egna-mobler-vid-datorn.html

Olsson, F. (1976). *Systematic Engineering Design (In Swedish. Original Title: Systematisk konstruktion)* (PhD Thesis). Department of Machine Design, Lund University, Lund.

Orsborn, S., Cagan, J., & Boatwright, P. (2008). Automating the creation of shape grammar rules. In J. S. Gero & A. K. Goel (Eds.), *3rd Design Computing and Cognition Conference DCC'08* (pp. 3–22). Dordrecht: Springer Science+Business Media.

Orsborn, S., Cagan, J., Pawlicki, R., & Smith, R. C. (2006). Creating cross-over vehicles: Defining and combining vehicle classes using shape grammars. *Artificial Intelligence for Engineering Design, Analysis and Manufacturing - AI EDAM*, *20*(3), 217–246.

Pahl, G., Beitz, W., Feldhusen, J., & Grote, K.-H. (2007). *Engineering Design - A Systematic Approach*. (K. M. Wallace & L. T. M. Blessing, Eds.) (3rd edition). London: Springer.

Pennell, J. P., & Winner, R. I. (1989). Concurrent engineering: practices and prospects. In *IEEE Global Telecommunications Conference, GLOBECOM'89* (pp. 647–655). Dallas, TX: IEEE.

Piaseki, M., & Hanna, S. (2010). A redefinition of the paradox of choice. In J. S. Gero (Ed.), *4th Design Computing and Cognition Conference - DCC'10* (pp. 347–366). Dordrecht: Springer.

Pine, B. J. (1993). *Mass Customization: The New Frontier in Business Competition*. Boston, MA: Harvard Business Press.

Prusinkiewicz, P., Hanan, J. S., & Lindenmayer, A. (1990). *The algorithmic beauty of plants*. Berlin: Springer.

Pugliese, M. J., & Cagan, J. (2002). Capturing a rebel: Modeling the Harley-Davidson brand through a motorcycle shape grammar. *Research in Engineering Design*, *13*(3), 139–156.

Reynolds, G. H. (2006). *An army of Davids - How markets and technology empower ordinary people to beat big media, big government, and other Goliaths*. Nashville, TN: Nelson Current.

Schoenauer, M., & Xanthakis, S. (1993). Constrained GA optimization. In S. Forrest (Ed.), *Proceedings of the Fifth International Conference on Genetic Algorithms* (pp. 473–580). Urbana, IL: Morgan Kaufmann.

Schwartz, B. (2004). *The Paradox of Choice: Why More is Less*. New York, NY: HarperCollins Publishers.

Shea, K., & Cagan, J. (1999). Languages and semantics of grammatical discrete structures. *Artificial Intelligence for Engineering Design, Analysis and Manufacturing - AI EDAM*, *13*(4), 241–251.

Simpson, T. W., Siddique, Z., & Jiao, J. R. (2006). *Product Platform and Product Family Design: Methods and Applications* (1st edition). New York, NY: Springer.

Svenska trä- och möbelindustrin (TMF). (2015). *TMF in Figures - Statistics on the Swedish Tree and Furniture Industry (In Swedish. Original title: TMF i siffror - Statistik om den svenska trä- och möbelindustrin).* Retrieved August 4 2015 from http://www.tmf.se/statistik/statistiska_publikationer/tmf_i_siffror.

Toffler, A. (1971). *Future Shock* (2nd edition). New York, NY: Bantam Books.

Ullman, D. G. (1997). *The Mechanical Design Process* (2nd edition). Singapore: McGraw-Hill.

Ulrich, K. T., & Eppinger, S. D. (2012). *Product Design and Development* (5th edition). London: McGraw-Hill.

Weber, F. (n.d.). Xylem [table]. Retrieved August 4, 2015, from http://www.behance.net/gallery/Xylem/355757

Welin, H. (2010, September 19). Nature is the inspiration (In Swedish. Original title: Naturen är förebild). *Sydsvenskan*, p. H3. Malmö: Bonnier.

Wenli, Z. (2008). Adaptive interactive evolutionary computation for active intent-oriented design. In *9th International Conference on Computer-Aided Industrial Design and Conceptual Design - CAIDCD'08* (pp. 274–279). Piscataway, NJ: IEEE.

Yeniay, Ö. (2005). Penalty function methods for constrained optimization with genetic algorithms. *Mathematical and Computational Applications*, *10*(1), 45–56.

Appended papers

Paper I

I

Exploration of the Domain of Application of the Renaissance 2.0 Approach

Nordin, A. & Hopf, A

Division of Machine Design, Division of Industrial Design,
Department of Design Sciences LTH, Lund University, Lund
2012

Exploration of the Domain of Application of the Renaissance 2.0 Approach

Axel Nordin and Andreas Hopf

Division of Machine Design • Division of Industrial Design

Department of Design Sciences

Faculty of Engineering LTH • Lund University • 2012

Exploration of the domain of application of the Renaissance 2.0 approach

Divisions of Machine Design and Industrial Design, Department of Design Sciences
Faculty of Engineering LTH, Lund University
Box 118
221 00 Lund

Abstract

The current form-giving activity in industrial design is, by and large, characterised by explorations that depend on an individual's capability to mentally manipulate a solution space from which to select and express the intended result. Designers often rely on artistic experimentation, aesthetic inspiration, or product specifications. Such approaches often lead to satisfactory results, but could profit from augmentation by methods of algorithmic form generation, optimisation, and use of natural-mathematical morphologies. By adopting *this approach,* designers would be able to efficiently use forms that have previously been too complex to handle and evaluate manually. If successful, such an approach would lead to the expansion of the morphological repertoire of the industrial designer, improve the integration of industrial design in the rest of the product development process, and, importantly, result in manifold opportunities for the customers to customise products.

The purpose of this report was to show that the **domain of application of the approach is large and, therefore, worth pursuing and further developing in future research**. To begin with, it was necessary to identify the key elements required for implementing the approach and then find members of each element category that, once combined, could facilitate the designing of very diverse products. These elements were: natural-mathematical morphologies, production technologies, materials and product typologies. Members of these element categories were then sought to ensure that a sufficiently large number of products could be designed through combinations of the element members. The conclusion is that such a feasible group of elements exists in the domain of 2.5D products based on 2D tessellations, sheet materials, and computer-controlled cutting and bending production technologies. Once combined, they enable creation and fabrication of very diverse products, even though only a small part of the domain has been surveyed. The conclusion is that the domain of application of the approach is vast.

Keywords: Computer aided design, morphological repertoire, industrial design, engineering design, production, generative design

Table of Contents

1 Introduction

1.1 The Renaissance 2.0 research programme

The Renaissance 2.0 research programme originated in search of new means for industrial designers to utilise natural-mathematical forms and structures, thereby extending their morphological repertoire. Preliminary work had been done by Andreas Hopf to categorise morphologies from animate and inanimate nature, depending on their origin and working principles. This work was extensive and was published in (Hopf 2009). One problem regarding the use of complex morphologies in the industrial design discipline is the multitude of constraints linked to the form-giving of products: surfaces are often functional, the artefacts are produced in several copies - meaning that the product form must accommodate available production technologies - and cost control is consequently important; finally, engineering constraints must also be respected. Therefore, enhancing the designer's morphological repertoire is tightly linked to the necessity to take technical constraints into account. From the consumer's point of view, there exists a growing desire to tailor products according to their personal preferences, rather than selecting from mass-produced offerings. If it is possible to enable an industrial designer to utilise new morphologies without having in-depth technical knowledge of the constraints linked to the products, it should also be possible to let customers directly influence form and function before purchase. These issues can be summarised in the overall goals of the Renaissance 2.0 research programme, which is to **enhance the morphological repertoire of industrial designers, improve the integration of industrial design in the product development process, and result in manifold opportunities for customers to customise products**.

1.2 The adopted approach

The approach adopted is that the goals of the Renaissance 2.0 research programme, mentioned above, can be achieved by developing a computer-based product design tool, which allows users without in-depth technical knowledge to design products while the tool takes technical and user constraints and objectives into account.

1.3 Purpose of the report and adopted approach

The purpose of this report **is to show that the domain of application of the approach is large** and thus worth pursuing and further developing in future research.

To achieve that goal, it is first necessary to identify the key elements of the approach, such as morphologies and production technologies, required for implementing the approach and then find a set of element members (such as using 2D tessellations as a morphology and laser cutting as a production technology) which, once combined, would allow for the realisation of a broad range of products.

1.4 Outline of the report

Chapter 2 presents constitutive elements of the approach and how they are related. Chapter 3 presents element members that enable the design of a broad range of products using the approach and their corresponding element members. Chapter 4 presents the conclusions from this study.

2 Elements of the approach and their interdependencies

Through the study of the elements of the product development process and elementary human-computer interaction models, a set of necessary elements of the approach has been derived, which are listed below (Section 2.1). Their interdependencies are shown in Section 2.2.

2.1 Elements of the approach

The key elements of the approach are described below and marked in bold. The ordering of the elements is not based on their importance but rather in the order they would appear in the interaction with the product design tool.

2.1.1 User

The **user** is the person who designs the product using the product design tool. The user can be an industrial designer internal to a company that uses the tool to design a single product or product family. It can also be a customer (consumers or businesses), or a sales person interacting with a customer.

2.1.2 Means of access

The **means of access** is the physical or virtual point of access where the user interacts with the product design tool. Depending on the user, this can be done in different locations. In the case of an industrial designer, the point of access could be through a software, used in-house, either individually or in cooperation with other colleagues. In the case of a customer, other possibilities emerge. The tool could be available online, either for an individual, or possibly for a community-based design process. The customer could also use the tool offline in consultation with a sales person – as is common when, for example, customising kitchen interiors.

2.1.3 Application platform

The **application platform** is the software environment in which the design tool is built. This depends on who the user will be. If the user is an industrial designer – with extensive experience in computer aided design tools software packages such as McNeel Rhinoceros® in conjunction with the graphical algorithm editor Grasshopper®, or Bentley® GenerativeComponents – integration in these software environments would be suitable. If, however, the user is a customer without extensive experience in 3D modelling, such systems might be daunting to use and not be economically feasible to adopt for a single or occasional use. In such cases, custom on- and offline applications built in Java or the open source Processing language would not require much effort in use on the customers' side and could be tailored to their needs and limited technical skills.

2.1.4 Interface

The **interface** of the product design tool is what the user interacts with to control the overall design and other variables such as **materials** and **objectives**. Obviously, one interface does not suit every user. While an industrial designer might feel most comfortable with a rather traditional 3D-modeling software interface with **full control** over all parameters, a customer not well acquainted with such software might prefer a dedicated solution with **limited control**, or one that allows for automated design generation.

2.1.5 Design generation

As in traditional product development, products designed with such toolkits must satisfy all technical **constraints** such as those linked to **production** and function. It should also be possible to **optimise** desired objectives such as the weight or cost of the product. Part of the aim of the Renaissance 2.0 approach is, however, that these technical constraints and objectives should be possible to handle even by non-technically oriented users. Depending on the **users' preferences** this could mean that the product design tool either handles constraints and optimisation automatically, or that information regarding how well the design performs in respect to the constraints and objectives is presented interactively to the users, enabling them to modify the design to suit. A third alternative would be to design the tool in such manner that no unfeasible designs can be generated, but this would require **algorithms** specific to every new product typology, morphology, and material combination to be used, which severely constrains the breadth of combinations that can be made available.

If the tool is to be able to automatically offer users design suggestions based on their preferences, it needs to **converge quickly** and **reliably** on solutions, and generate **diverse** designs. As noted earlier, the design generation system should be **generic** in nature to be able to handle a wide range of products without requiring substantial modification to the design generation system itself. The design generation can be based on computational search methods, which try to satisfy constraints and optimise objectives, presenting only feasible solutions to the user. In this case, it is important that the design generation system can offer a diverse range of design suggestions to the user so that their choice is not limited.

2.1.6 Morphologies

The reservoir of forms and structures, compiled by Andreas Hopf (2009) , is based on natural-mathematical **morphologies**. The morphological repertoire should be thought of as a source of inspiration; as starting point for a new design or a reference for finding suitable forms and structures for an existing idea. Many natural-mathematical morphologies, although interesting from an aesthetic and functional point of view, are complex to design with and materialise. Not all morphologies are feasible to be put into production unless rapid prototyping technologies are used, because they can be very intricate – and not all morphologies are suitable for every type of product.

2.1.7 Product typologies

The **product typologies** suitable for the Renaissance 2.0 approach are dependent on the feasible subset of morphologies, the technical complexity of the product, and the intended target group for the product. There might be products, which are too complex, at least given today's simulation technology, to be investigated and analysed solely digitally. If prototypes must be built to verify the function and technical feasibility of the design, many benefits of using an integrated product design tool disappear. If the product is only to be produced as a limited edition or one-off, the overhead cost of developing an integrated product design tool might not be justified.

2.1.8 Production technologies

If customers are allowed to customise the product form, it is necessary to adopt more flexible **production technologies**, with which to produce individualised products. There exist many such technologies, such as computer numerically controlled (CNC) and rapid prototyping machines, which can produce 100 individualised objects at the cost of 100 identical ones. However, it is also necessary to consider the step before the actual production, the **production preparation**. The time required for translating the output of the design tool to machine instructions, and the time needed to set up the machines before producing a new part is small in comparison to the actual production time for large batch runs, but for individualised products these added costs become a factor. The production technologies used to realise such products are of course constrained by the chosen morphology, material, and product typology, but also by the adopted business model. If an industrial designer were to use the product design tool to develop conventional products in-house, the obvious solution would be to use traditional mass production technologies.

2.1.9 Business models

The scope of application of new digital means of interaction, designing and production is fully scalable and in that sense constitutes a unique enabler that, if consistently implemented, potentially cuts across a very large number of industries – ranging from small manufacturers to large producers of consumer products.

Various entrepreneurial opportunities arise on the basis of the Renaissance 2.0 approach:

1. A design studio – designing for established manufacturers that sell to their customers.
2. A design studio – designing and selling directly to customers online; the products necessitating a **supply chain**.
3. A design studio "work-shop" – designing and selling directly to customers from a "work-shop", augmented with an online presence; the products being produced by a network of suppliers.
4. A design enabler – customers (co-)design under the companies' tutelage, their products being produced by a network of suppliers.

5. A software consultancy – consulting established manufacturers in enhancing the capabilities of their internal design studio – or manufacturers wanting to add co-creation strategies to their existing business model.

2.2 Elements' interdependencies

Interdependencies between the elements of the approach are presented in Figure 2.1.

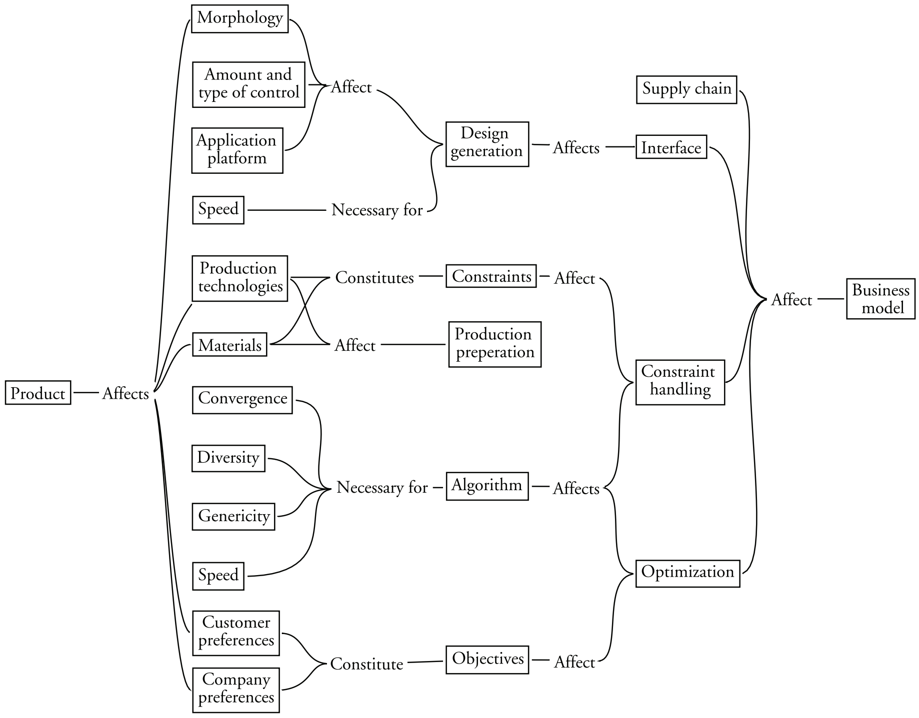


Figure 2.1. Network of interdependencies of the elements of the approach

3 Identification and extent of the elements which enable the design of very diverse products

First, the subset of elements that enable the designing of very diverse products using the approach is presented in Section 3.1. Then their corresponding element members are listed (Section 3.3 to 3.6). Section 3.2 presents the adopted approach to survey those element members.

3.1 Identification of the elements which enable the designing of very diverse products using the approach

As summarised in Figure 2.1, there exists a network of interdependencies between the different elements of the Renaissance 2.0 approach. The identification of these dependencies is useful for structuring a research strategy to test the feasibility of the entire framework. Nevertheless, for determining the domain of application of the approach, only a subset of those elements needs to be further investigated.

It is evident that:

- morphology,
- production technology,
- material, and
- product typology

are closely intertwined, and cannot be selected separately; one must always consider the materials used when deciding on the production technology and vice versa. The range of the products that can be designed using the approach is therefore dependent on the breadth of the following elements: “morphologies”, “production technologies”, and “materials”, and on their possible combinations. On the other hand, the range of those products is not systematically constrained by the other elements. These elements are surveyed in the Sections 3.3 to 3.6.

3.2 Approach

As mentioned above, the range of the products that can be designed by the approach is constrained by the identified elements and their possible combinations. It would be very difficult and time-consuming to be exhaustive in the inventory and combination of all element members. This has been done for the morphologies (cf. Section 1.3) but for the other elements another approach was adopted. It was decided to search for element members which 1) work well together and enable the generation of many products while

2) being inexpensive and readily available, but still 3) posing the same type of challenges more complex technologies would, such as structural stability, producibility, and overall functionality. These considerations led to the decision to look into combinations of 2.5D products based on 2D morphologies, sheet materials and corresponding compatible production technologies as they should be relatively straightforward to produce, yet still enable the output of very diverse product typologies.

The morphologies, the production technologies and materials are described in Sections 3.3, 3.4 and 3.5 respectively. Compatible 2.5D products are presented in Section 3.6.

3.3 Morphologies

Since the long term objective is to research the aesthetic and functional dimensions of algorithmic and evolutionary form generation, we have investigated a substantial range of natural-mathematical morphologies, although only a subset is used in this initial study. The result of this investigation is summarised in Figure 3.1 and Figure 3.3, with examples of the morphologies shown in Figure 3.2 and Figure 3.4. Figure 3.1 presents the classification of different morphologies according to their geometries. Figure 3.3 links the different morphologies with nature and how their use can be exploited as a source of inspiration (both for aesthetic and functional purposes).

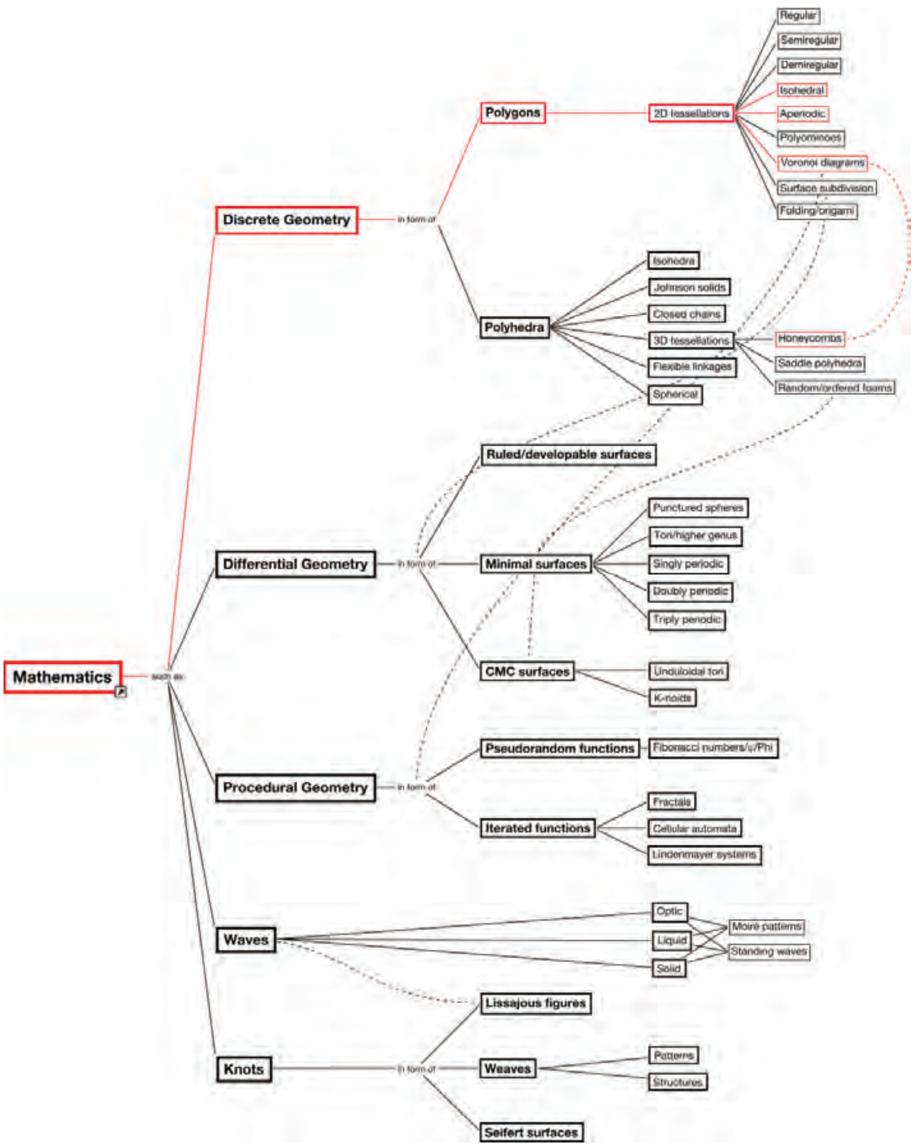


Figure 3.1. Partial representations of morphologies derived from mathematics (from (Hopf 2009, p. 11))

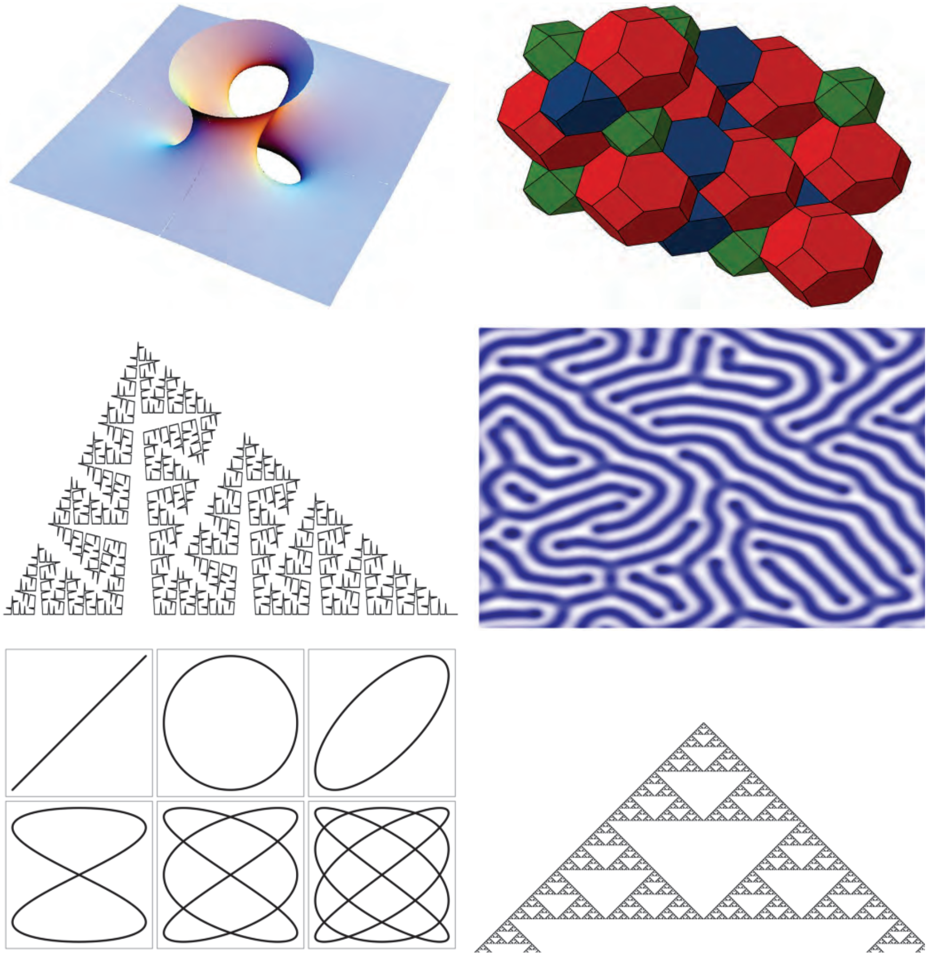


Figure 3.2. Examples of mathematical morphologies. From top left to bottom right: minimal surface[1], space filling polyhedra, Lindenmayer system, reaction diffusion system[2], Lissajous graph, cellular automata[3]

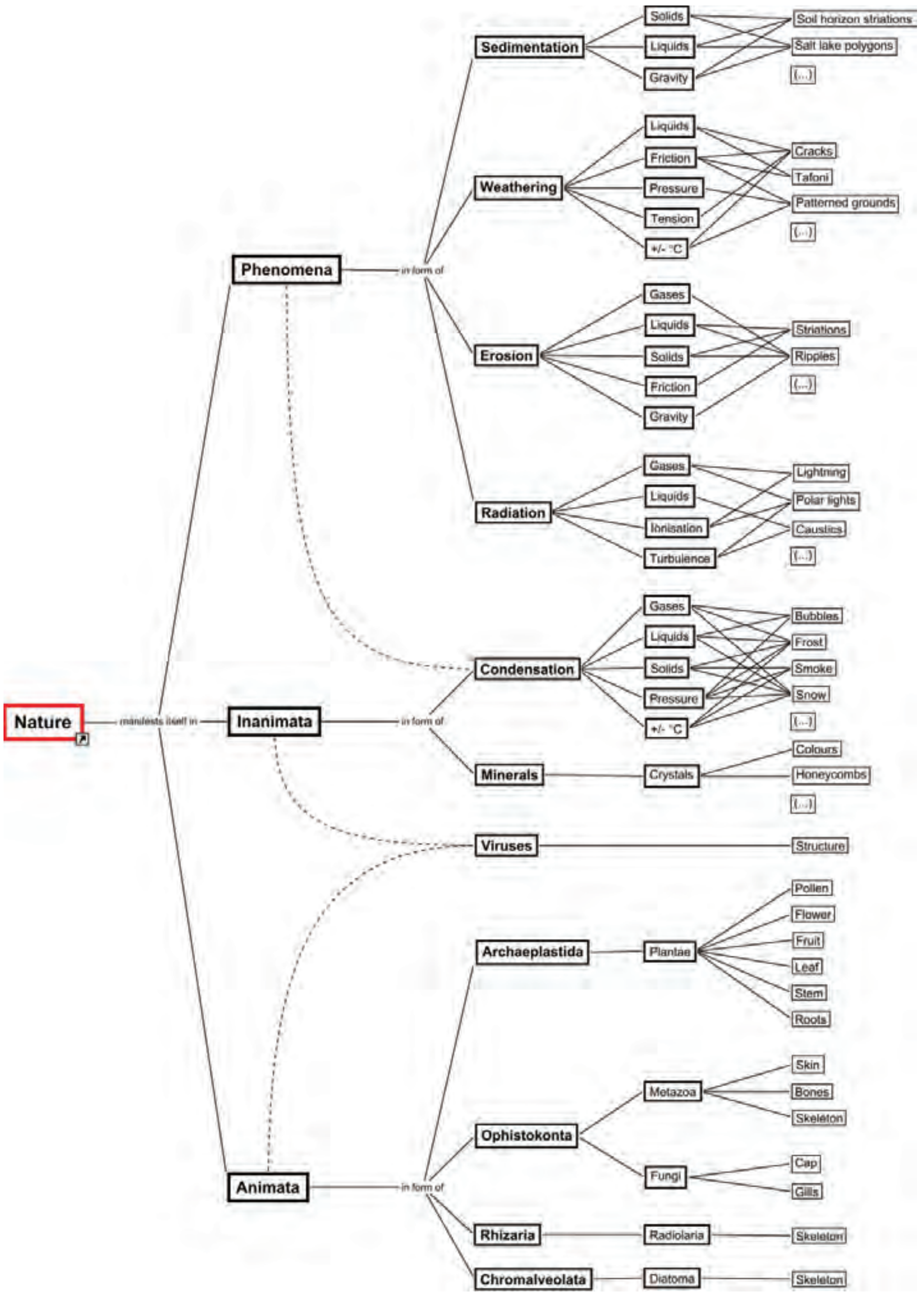


Figure 3.3. Partial representations of morphologies originating from nature (from (Hopf 2009, p. 9))

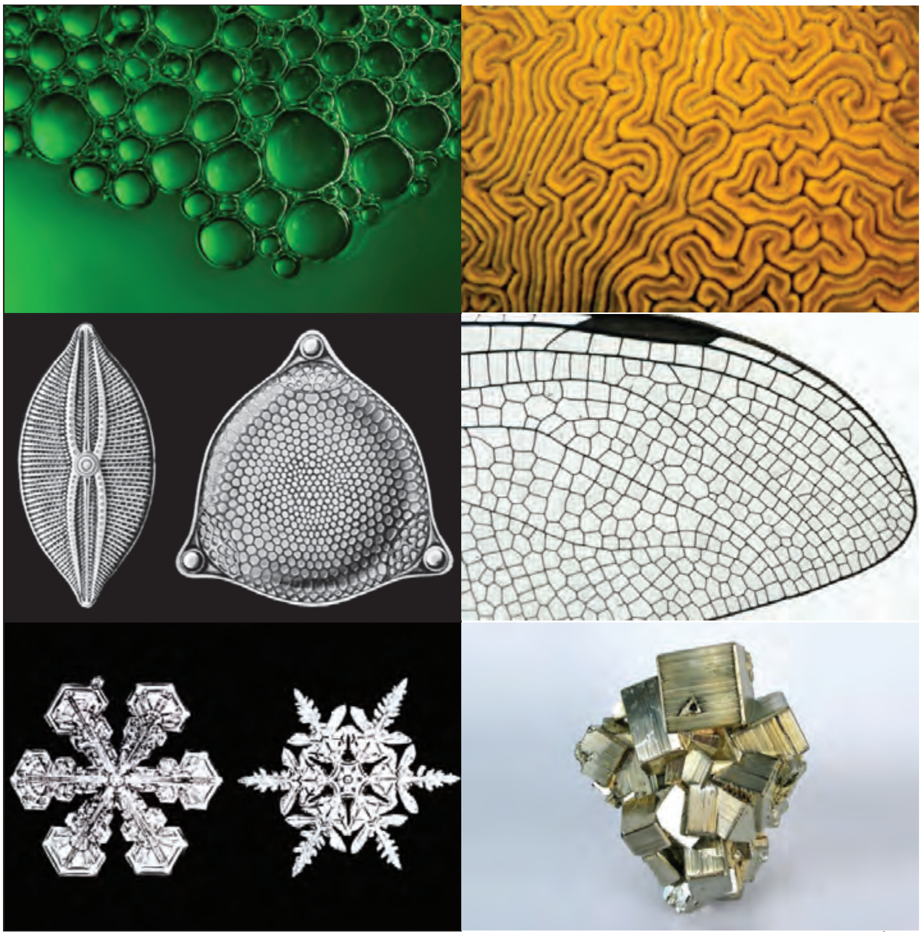


Figure 3.4. Examples of natural morphologies. From top left to bottom right: Soap bubbles[4], brain coral[5], diatoma[6], dragonfly wing[7], snowflakes[8], pyrite crystal[9]

[4] © 2007 woodleywonderworks / CC BY-SA 3.0, available at http://www.flickr.com/photos/wwworks/667298782/
[5] © 2010 Nick Hobgood / CC BY-SA 3.0, available at http://commons.wikimedia.org/wiki/File:Diplora_labyrinthiformis_(Grooved_Brain_Coral)_closeup.jpg
[6] Ernst Haeckel., *Kunstformen der Natur*.
[7] © 2009 Jerry Porsbjer / CC BY-SA 3.0, available at http://commons.wikimedia.org/wiki/File:Sl%C3%A4ndvinge_01.jpg
[8] Wilson Bentley, *Studies among the Snow Crystals*
[9] © 2007 Robert Lavinsky / CC BY-SA 3.0, available at http://commons.wikimedia.org/wiki/File:Pyrite-131884.jpg

Out of the many morphologies, 2D tessellations were explored in depth as there is a vast repertoire of them, both complex and simple, they are straightforward to visualise, and can be applied to 2.5D objects, using many available production technologies. They are also suited to very different product typologies as will be shown later. In particular, the Voronoi tessellations (which themselves are a superset of various other regular tessellations), three isohedral tessellations (D1 pentagon, D2 hexagon, kite) and one aperiodic tessellation (Chinese lattice or ice-ray lattice) were further investigated.

3.3.1 The Voronoi diagram

A Voronoi (or Thiessen) structure is shown in Figure 3.5. Phenomena as diverse as the wing of a dragonfly, the structure of bone marrow, and a honeycomb can be described with Voronoi diagrams (Aurenhammer 1991). Such structures are often found in both lightweight yet strong structures in nature (Beukers and van Hinte 2005; Pearce 1978). Apart from aesthetic aspects, a Voronoi-based bookshelf would consequently have a structure well suited for carrying heavy loads, such as books and magazines, whilst maintaining a low weight. A Voronoi diagram can be described as follows: Let $p_1,.., p_n$ be a set of n distinct points in the plane; these points are called the Voronoi sites. For each site, the set of points that are closer to it than to any other site form a Voronoi cell. A Voronoi diagram is constituted of all such cells. An overview of a Voronoi diagrams' properties can be found in (de Berg et al. 2008, chapter 7). The Voronoi diagram provides many variables in the form of the Voronoi sites, which can be positioned by the user. A small change in the position of one of the sites can lead to large changes in the entire Voronoi diagram, which makes it an interesting and challenging morphology to control, and is suited for the study of how well complex morphologies can be handled in an interactive product design tool.

3.3.2 The Chinese lattice tessellation

The Chinese lattice can be thought of as a dynamic shape grammar, the structure is basically generated by bisecting any polygon (see Figure 3.5). A thorough description of the generation of Chinese lattices can be found in (Stiny 1977). As with the Voronoi diagram, there exists the possibility to create a wide range of structures by manipulating the control points for the Chinese lattice tessellation, the resulting aesthetic is, however, very different from the somewhat organic looking Voronoi diagram, which makes it suitable to include in studies to investigate how much the user's experience with the tool depends on a morphology's aesthetic.

3.3.3 Isohedral tessellations

The isohedral tessellations used in the application are two-dimensional and tile the Euclidian plane. An isohedral tiling consists of polygons surrounded by copies of themselves. There exist 42 unique isohedral tessellations consisting of symmetric polygons. Three of these have been implemented in the application, the pentagonal D1, the hexagonal D1, and the kite tessellation. See Figure 3.5. An in-depth description of isohedral tessellations can be found in (Schattschneider and Dolbilin 1998). These tessellations are very simple to control, at most there are two control points which can be re-positioned, and the tessellations are built from one polygon which is repeated, thus making the morphology easy to understand and control. These tessellations are interesting to include in future studies together with the more complex Voronoi diagram and Chinese lattice tessellation to investigate how the complexity of the morphology influences the user's experience.

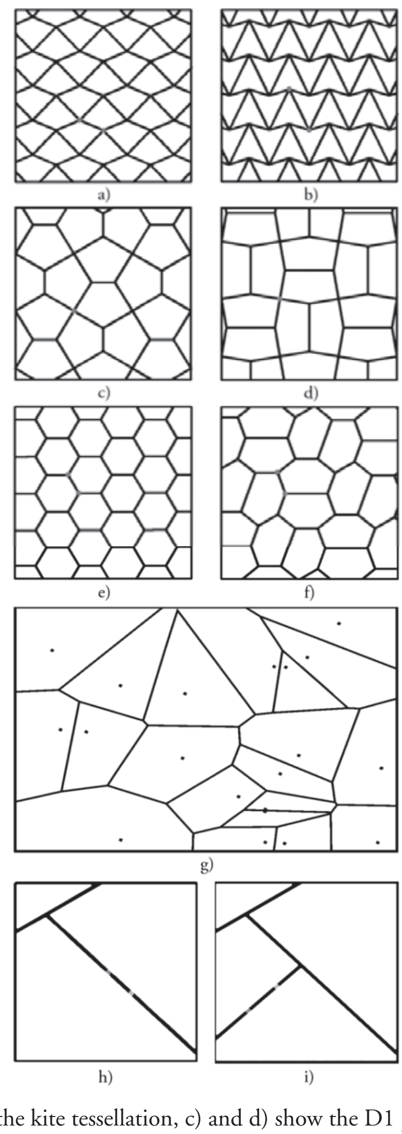


Figure 3.5. a) and b) show the kite tessellation, c) and d) show the D1 pentagonal tessellation, e) and f) show the D1 hexagonal tessellation, g) shows the Voronoi tessellation, and h) and i) show the Chinese lattice tessellation. The control points for the isohedral tessellations and the Chinese-lattices are marked in grey.

3.4 Production technologies

An important observation is, that, in many cases of experimental computational form generation, rapid prototyping is the fabrication system of choice, c.f. (Wertel and Oberfell 2007) and (Joris Laarman 2010). Restricting the application of an extended morphologic repertoire to rapid prototyping may not be sustainable in the long term, because only very few types of products are actually suitable for this fabrication technology. Rapid prototyping is likely not to be the panacea; it still suffers from various disadvantages such as limited build space, long build times, non-heterogeneous materials. That is why we focused on "traditional" digital fabrication technologies such as laser cutting and CNC sheet metal bending technologies.

Although manufacturing constraints can be respected in the design process, it is necessary to find manufacturing technologies allowing for one-off production (bespoke products) at a reasonable cost and within a reasonable time frame. Historically, all products were bespoke, and in certain fields such as bespoke clothing, shoes, and interior architecture this is still common, although an abundance of low cost mass-produced alternatives exist. However, the cost and waiting time for such bespoke products is still prohibiting average consumers from purchasing products tailored to their preferences. With the rapid development of computer controlled production processes, the cost of producing one item is converging on the cost of producing one thousand. However, the cost of ensuring that all technical constraints have been satisfied and preparing the machine instructions is still obstructing true "mass-one-off" production. Recent advances in rapid-prototyping are making the production of small scale products of low complexity and with few engineering constraints feasible, but larger products with demands on structural integrity and complex assemblies are still too costly and time-consuming to produce with rapid prototyping technology. The question is therefore, how can well-established computer controlled manufacturing technologies be efficiently adapted to the production of "mass-one-offs". Therefore, one needs to investigate, how the control of production and engineering constraints and the process of translating the digital forms into instructions for efficient production of physical objects can be managed.

The production technology of a product is intimately tied to its form and function. It constrains the possible materials, dimensions, tolerances, variability, and shapes. A small overview of the current production technologies suitable for customisable products will be given here, with some advantages and disadvantages. An overview of the computer controlled production processes can be found in Figure 3.6. More in-depth information about these processes can be found in (Kalpakjian and Schmid 2010) and elsewhere.

3.4.1 Casting processes

Although requiring moulds, casting processes such as sand casting or slip casting are suited for small batch production at a relatively low cost, if CNC-milling or rapid prototyping is used for mould fabrication. Compared to the direct production of parts with CNC-milling or rapid prototyping, the main advantages are the reusability of the moulds to create multiple objects, the use of materials which are not feasible for milling, such as glass or ceramics, and the improved surface quality, strength, and cost compared to rapid prototyping.

3.4.2 Sheet material processes

The use of computer controlled bending, cutting, and spinning makes sheet material processes suitable for direct fabrication of customisable objects. Processes such as laser, water-jet, and plasma cutting make it possible to process a wide range of materials such as metals, ceramics, and woods. Thin-walled objects with high strength, good surface quality, and a large selection of materials are the main advantages of these processes. The drawbacks consist mainly of the complexity in finding bending and joining operations that will result in the desired shape and function.

3.4.3 Polymer processing

Polymer sheet materials can be processed using the sheet material processes, and, in conjunction to those, also vacuum formed, which is a cost-effective alternative for the production of limited batches where the mould can be made out of low cost materials such as medium density fibreboard as the structural requirements of the mould in vacuum forming is relatively low. In addition to traditional processes many rapid prototyping technologies use polymers as the main material, such as stereolithography, fused deposition modelling, and 3D printing. As noted earlier, the rapid prototyping technologies' main disadvantages are the high cost, slow manufacturing, and low surface quality. They are, however, becoming more common for products that are small, have few structural requirements, and have intricate shapes that would be difficult to produce by other means.

3.4.4 Machining processes

Most of the traditional machining processes such as turning, drilling, and milling have been computerised since the 1960s. The flexibility of machining is determined by the number of controllable axes of the machine, in many cases a standard three-axis machine is sufficient for most operations, but more advanced machines can move in five or more axes – enabling undercuts and a higher surface quality. Originally, CNC-machines were used to manufacture tools for casting, stamping, and drawing as the cost and speed of the process was prohibiting mass production, but recently they have begun to be used extensively for mass production as well. CNC-machines can still not compete with traditional mass production processes such as injection moulding or extrusion, but they offer very good surface quality and high precision. The main challenge in using CNC-machines for customisable products is the production preparation phase where the 3D model to be produced is translated into operational code for the machine, while ensuring that all manufacturing constraints are satisfied, a process which can be time-consuming and is a substantial part of the total cost of production, if only a few exemplars are produced.

3.4.5 Joining processes

Through proliferation of industrial robots during the last 40 years, many joining processes that were previously executed manually such as welding, gluing, etc. have been automated. Industrial robots are also used extensively in the automotive industry for the painting and finishing of car exteriors.

The development seen since the 1980s enables the entire process from cutting, machining, joining, painting, finishing, and assembly to be fully automated. The process of creating manufacturing instructions and synchronising the entire workflow still remains a time consuming and difficult task, which makes fully customisable products not yet viable, given the current technology.

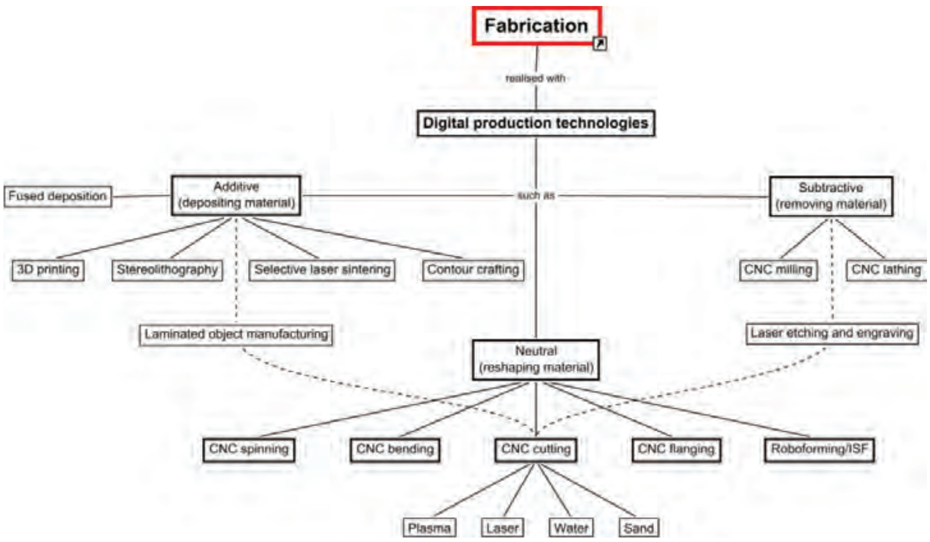


Figure 3.6 An overview of some available computer controlled manufacturing processes

3.5 Materials

Given the element technologies suggested are applied to 2.5D product typologies based on 2D tessellations, the use of sheet materials in the form of sheet metal, sheet wood, float glass, ceramics, and plastics is obvious. Sheet materials are well suited for the production of 2.5D objects via creating cells extruded from the 2D tessellations. Very little material is wasted compared to milling products or product components from solid material, and many production technologies exist for the cutting and forming sheet materials which are relatively inexpensive.

3.6 Product typologies

Very diverse products can be developed adopting the proposed approach. Figure 3.7 shows a broad range of products for which the design can be based on 2D tessellations (see Figure 3.1), combined with the suggested production technologies and materials. Many more can be envisaged using other morphologies and other production technologies. These possibilities are waiting to be explored.

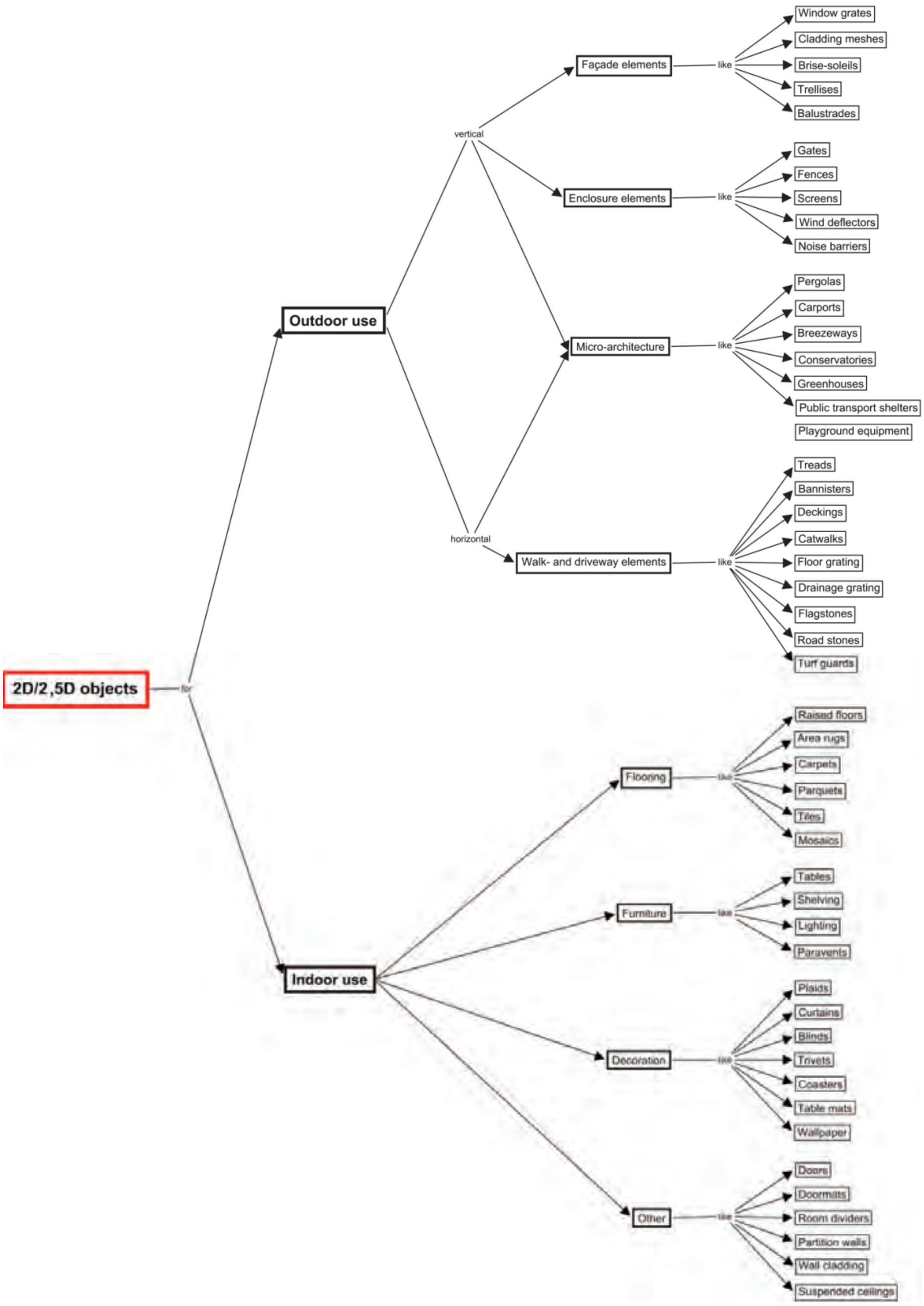


Figure 3.7 An overview of various 2.5D products

4 Conclusion

This report introduces the Renaissance 2.0 research programme. It aims to expand the industrial designer's morphological repertoire, to advance the integration of industrial design in the product development process, and to enable customers to customise a broad range of products. The approach adopted for achieving the aims of the programme is the development of a computer-based product design tool, which would allow users without in-depth technical knowledge to design products, whilst automatically taking into account technical and user constraints and objectives. The purpose of this report has been to show that the domain of application of the approach is very large. To study the extent of the approach, it was first necessary to identify the key elements of the approach, and then to find suitable element members for each element that enables the designing of a broad range of products.

The study resulted in a list of necessary elements and their corresponding element members that enable the approach. The required elements for the approach, as well as their interdependencies, are shown in Figure 2.1.

The key elements that enable the designing of products using the approach are product typologies, morphologies, production technologies, and materials. It could be shown that there exists a large set of products, that is, a set of combinations of product typologies, morphologies, production technologies, and materials, which are possible to design for with the approach. The study shows that one can identify at least one feasible set of element members enabling the approach. This subset consists of 2.5D products such as bookshelves, room dividers, tables and many more, designed based on 2D tessellations such as the Voronoi-diagram, or isohedral tessellations, produced from flat sheet materials such as plywood and stainless steel, with CNC production technologies such as laser cutting, water jet cutting, and bending. These element members alone enable a wide range of feasible combinations, which is a strong argument in favour of the relevance of the approach. Future studies should implement the approach and test its technical and economic feasibility, as well as surveying other applicable element members to further expand the domain of application of the approach.

References

Aurenhammer, F., 1991, "Voronoi diagrams - A survey of a fundamental geometric data". *ACM Computing Surveys*, *23*(3), pp. 345-406.

Beukers, A., van Hinte, E., 2005, *Lightness: the Inevitable Renaissance of Minimum Energy Structures* (4th Revised Edition). 010 publishers, Rotterdam.

de Berg, M., Cheong, O., van Kreveld, M., Overmars, M., 2008, *Computational Geometry - Algorithms and Applications* (3rd Edition). Springer-Verlag, Berlin.

Hopf, A., 2009, "Renaissance 2.0 - Expanding the morphologic repertoire in design". *24th Cumulus Conference 24/09*, Melbourne, November 12-14, 2009, pp. 78-85.

Joris Laarman, 2010, "Starlings". <http://www.jorislaarman.com/particles.html>2010, last accessed: 1 May 2012.

Kalpakjian, S., Schmid, S. R., 2010, *Manufacturing Engineering and Technology* (6th Edition). Prentice Hall, Upper Saddle River, NJ.

Pearce, P., 1978, *Structure in Nature is a Strategy for Design*. MIT Press, Cambridge, MA.

Schattschneider, D., Dolbilin, N., 1998, "One corona is enough for the Euclidean plane". In Patela, J. (Ed.), 1998, *Quasicrystals and Discrete Geometry*, American Mathematical Society, Providence, RI, pp. 207-246.

Stiny, G., 1977, "Ice-ray: A note on the generation of Chinese lattice designs". *Environment and Planning B*, *4*(1), pp. 89-98.

Wertel, J., Oberfell, G., 2007, *Fractal-T* [Table]. In Klanten, R., Ehmann, S., Kupetz, A., Moreno, S., Mollard, A. (Eds.) (Eds), *Desire - The Shape of Things to come*, Die Gestalten Verlag, Berlin, 2008, http://www.materialise.com/materialise/view/en/2560204-Platform+Studio.html, last accessed: 2009-09-23.

Paper II

II

An approach to constraint-based and mass-customizable product design

Nordin, A., Hopf, A., Motte, D., Bjärnemo, R.,
& Eckhardt, C.-C.

Journal of Computing and Information Science in Engineering

Originally published by ASME

Axel Nordin
Machine Design LTH,
Lund University,
SE-221 00 Lund, Sweden
e-mail: axel.nordin@mkon.lth.se

Andreas Hopf
Industrial Design LTH,
Lund University,
SE-221 00 Lund, Sweden
e-mail: andreas.hopf@design.lth.se

Damien Motte
e-mail: damien.motte@mkon.lth.se

Robert Bjärnemo
Mem. ASME
e-mail: robert.bjarnemo@mkon.lth.se

Machine Design LTH,
Lund University,
SE-221 00 Lund, Sweden

Claus-Christian Eckhardt
Industrial Design LTH,
Lund University,
SE-221 00 Lund, Sweden
e-mail: claus.eckhardt@design.lth.se

An Approach to Constraint-Based and Mass-Customizable Product Design

In traditional product development, several iterations are usually necessary to obtain a successful compromise between constraints emanating from engineering, manufacturing, and aesthetics. Moreover, this approach to product development is not well suited for true mass-customization, as the manufacturing company remains in control of all aspects of the shape of the product-to-be. In this article, we propose an alternative approach that would (1) allow for an improved integration of industrial design into the product development process and (2) enhance the creative repertoire of industrial designers, which (3) would result in significantly improved prospects for mass-customization. The industrial design process may benefit from using advanced and aesthetically interesting morphologies emanating from the areas of mathematics and nature. Such complex morphologies can only be manipulated (analyzed and represented) by means of specific algorithms. On one hand, this requires a shift from established industrial design practice, where the industrial designer is in total control of the product form; on the other hand, it makes it fully possible to compute form so that it complies with engineering and manufacturing constraints. In this setup, the industrial designer still has control of the final result, in that she or he can choose from a set of valid forms. This approach would greatly reduce the number of iterations in the product development process between industrial design, engineering, and production. Naturally, such an approach also allows for advanced mass-customization by allowing consumers to use these tools. Within this approach, a table generation system has been developed: A system that generates tables whose support structure is based on a Voronoi diagram that fulfills structural and manufacturing constraints while being aesthetically appealing. [DOI: 10.1115/1.3569828]

1 Introduction

In the traditional product development process, several iterations are usually needed to obtain a compromise between engineering and manufacturing constraints on the one hand and aesthetics on the other. The process typically begins with a design brief to which the industrial designer (hereafter, "designer") develops initial concepts, which are reviewed and decided on in close collaboration with marketing, engineering design, and production departments. The chosen concept is refined until it satisfies the engineering and manufacturing requirements and can be produced. This traditional approach limits severely any subsequent option for mass-customization related to the design of the product: An option to which customers are becoming more attuned. Finally, while the new digital means of creation and fabrication empower designers with new levels of freedom to even become market actors in their own right, they are rarely educated to take full advantage of the new methodologies of creation, especially regarding the competence to exploit the extraordinary reservoir of morphologies from nature and mathematics, as has been the case within the area of architecture for quite a while now. In this article, we want to propose an alternative approach that would (1) allow for an increased integration of industrial design (hereafter, "design") into the product development process and (2) enhance the creative repertoire of the designer, which would eventually (3) result in significantly improved prospects for mass-customization.

Today, many natural and mathematical structures can be computed, analyzed, and graphically represented in a manageable time. It is believed that designers could use specific tools in order to create concepts and designs that they could never have conceived without the support of such tools. The new morphologies can then be optimized with reference to customers' specifications and engineering and manufacturing constraints while submitted to the critical judgment of the designer. This approach has the potential to allow active customer participation, which might even result in the actual origination of the design in the spirit of true mass-customization.

The main focus of this article is to elaborate on the feasibility of such an approach. Can the customers' specifications as well as the engineering and manufacturing constraints be taken into account in a semi-automated design process without compromising aesthetics? To test this approach, a table generation system has been developed. The designer *or* end customer (hereafter, user) has some degrees of freedom concerning the form of tables. A 2D Voronoi diagram, see Fig. 1(*a*), was chosen as a novel tessellation for the supporting parts of tables and was submitted to a set of structural and manufacturing constraints. The developed system allows for the search of tables that fulfill those constraints while being aesthetically appealing. The first section of this article develops this approach in some depth and compares it to related works. Then, the heuristic used and the application developed are presented.

2 Background

Design is by and largely considered an integral part of most product development processes. Design activities are performed either by external designers (design agencies or freelance designers) or by an internal design department. In the first case, design is somewhat in the margin of product development: External designers are hired for one project, they often work relatively isolated and their proposed concepts will inevitably result in further dis-

Contributed by the Computational Geometry/Visualization Committee of ASME for publication in the Journal of Computing and Information Science in Engineering. Manuscript received October 12, 2009; final manuscript received February 25, 2011; published online March 31, 2011. Assoc. Editor: S. McMains.

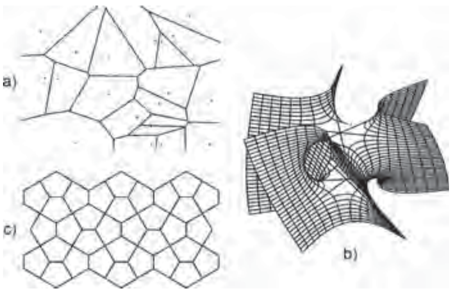


Fig. 1 Examples of (*a*) a Voronoi diagram, (*b*) a minimal surface (twisted Scherk surface), and (*c*) a D1-tessellation

cussions in terms of engineering and manufacturing constraints. In the second case, even when there is extensive collaboration between the different functions of the company, iterations are still necessary. For the majority of companies, the development time is of utmost importance and a reduction of the number of iterations is a priority.

The traditional product development process also puts restrictions on the possibilities for mass-customization. Customers may occasionally decide between different colors or materials but it is almost impossible to allow them to generate a form of their own choice, apart from the possibility to select predefined product variants and models.

The designer is rarely educated to make conceptual use of complex forms such as those derived from Voronoi diagrams or minimal surfaces (Fig. 1). This is partly due to the fact that those forms generally require advanced knowledge in mathematics but more so due to the idea that the designer must be an intuitive *author*; hence, often disregarding morphologies that would require the expertise of others. This dilemma was discussed in length in Ref. [1].

3 Related Work

The association between design and related topics, such as new morphologies, mass-customization, and integration of engineering and manufacturing constraints, has already been considered in literature but mostly as discrete issues.

In academia, expanding the creative repertoire by using morphologies from nature and mathematics has been the concern of design research for some time now, see, e.g., Ref. [2], although such ideas have been deployed mainly within architecture [3,4] or graphic design. This concern has also been elaborated within engineering design [5,6] and in the development of materials [7,8]. More recently, McCormack et al. [9] presented a model of what they call generative design and proposed an appropriate agenda for design research, although the examples were not taken from the area of industrial design. The research of generative design in industrial design concerns primarily the (1) generation of artifacts based on a particular *style* [10,11] (2) and *branding* related issues [12–14]. Shea and Cagan [15] used a combination of shape grammar and simulated annealing for both functional and aesthetic purposes and applied it to truss structures. In Ref. [15], the aesthetical constraints were formalized by using the properties of the golden number. Shea and Cagan's [15] approach is the most similar to ours. In our case, though, it is necessary to let users decide on their own preferences. This is achieved by using an interactive method.

Some applications can be found in industry, mainly developed by design agencies and leading designers. For example, Trubridge's ceiling lamp is based on classic polyhedral geometry [16]. In general, for products with few structural constraints, such as lamps, mathematical algorithms have been used to develop new forms: This permits to focus on the aesthetic characteristics while the remaining constraints can be neglected. These structures have also been used together with rapid prototyping, permitting the production of virtually every form but is an expensive and time-consuming production system: Wertel and Oberfell from Platform Studio (Leuven, Belgium) developed a mineral-based table, Fractal-T [17], manufactured with rapid prototyping. Moreover, many designs remain concepts: They are presented at design fairs and showrooms, with the purpose of showing the aesthetical and functional potential of complex structures and inspire other designers rather than making a commercial exploitation of them. For example, van der Veer developed a paper table, demonstrating the strength of paper in combination with mathematical models [18], and Mayor's burnout bench is based on a sculptural wave described in 3D by a computer [19]. In the automotive industry, Mercedes has developed a concept car based on the properties of the boxfish, "respecting at once physics, design, and aerodynamics" [20]. In comparison to industrial design, there are numerous examples of algorithmically generated and optimized architecture: London Town Hall, London Swiss Re Building, etc. Nevertheless, the constraints are different. Since a building is most often a one-off product, once a proposal has been accepted at the conceptual level, the architect is assured to receive financing for "manufacturing." The risks associated with the project are thus more or less eliminated.

Focus on form has usually been decoupled from the other aspects commended in this project: mass-customization and integration of production constraints in the design activity. Concerning the latter, many methodologies and tools have been developed for the integration of engineering and manufacture [21] but industrial design has been neglected. In the domains of product platform and product family design, the algorithms developed to automate the generation of product variants and models, e.g., Refs. [22,23], take engineering and production constraints into account but not aesthetics. DASSAULT SYSTÈMES CATIA® and PTC's (Needham, MA) PROENGINEER® each have implemented a module that permits freeform design in a format compatible with their computer-aided design (CAD) system: Imagine and Shape and Pro/Concept, respectively. Such plug-ins can accelerate the design process and represent a step toward increased integration but still do not implement any extended morphological repertoire, and the engineering and manufacturing constraints are treated after the first concepts have been produced.

Concerning advanced mass-customization, where the user has a direct influence, graphic design has experienced some success stories, where customers are completely free to design whatever motive they want, as shown by the Harvard Business School case reported in Ref. [24]. These are among the examples described by von Hippel [25,26], who initiated the user innovation paradigm. In electronics and software industry, mass-customization is also highly present. The iPhone® from Apple is no longer primarily considered a mobile phone with a user-friendly interface but as a platform in which everyone can build on. Once again, however, the production constraints are not identical [27]. The novelty of this project is thus to bring together these three aspects, new morphologies, mass-customization and integration of design, and engineering and production, which should, in a long term perspective if successful, reduce time-to-market and costs, and give a competitive advantage to those companies implementing it.

4 Approach and Application

In this paper, we propose an approach that can be appropriate when aesthetical aspects and mass-customization are prioritized. Many morphologies from nature (which has long been a source of inspiration in design) have been described in a computational form. In geometry, many shapes have been developed that have structural properties (for example, minimal surfaces, Fig. 1(*b*)) and are aesthetically remarkable. By coupling such computer-based morphologies to engineering and manufacturing constraints, it is possible to generate forms that are both structurally sound and visually appealing. Advances in optimization methods and artificial intelligence make this integration possible. At the same time, this approach facilitates the development of variants that can be tailored to each customer (individuals or groups). It also allows for the designer or the customer (the user) to intervene during the optimization process (interaction method). Letting the customer to *co-create* the final product makes a truly mass-customized production system a reality.

This approach is detailed in Sec. 4.1. The approach is illustrated by an application presented in the next sections.

4.1 The Approach. The class of problems concerned by our approach has the following characteristics: (1) a complex morphology is integrated in a product, (2) the user can partially control the features of the morphology in order to make the product unique, (3) the morphology must comply with engineering and manufacturing constraints, and (4) the complex morphology affects the product's functionality and aesthetics. The third characteristic implies that the morphology is not just decorative but has some structural function; it implies also that the approach takes into account manufacturing systems other than just rapid prototyping.

In this general case, then, some functional and aesthetic engineering and manufacturing objectives $\mathbf{F}(\mathbf{x})$ need to be minimized or maximized (for example, costs and weight) while other functional and aesthetic engineering and manufacturing conditions are constraints $\mathbf{G}(\mathbf{x}) \leq \mathbf{0}$ and $\mathbf{H}(\mathbf{x}) = \mathbf{0}$. This is a problem of multiobjective optimization

$$\begin{aligned} &\text{minimize} \quad \mathbf{F}(\mathbf{x}) \\ &\text{subject to} \quad \mathbf{G}(\mathbf{x}) \leq \mathbf{0} \quad \text{and} \quad \mathbf{H}(\mathbf{x}) = \mathbf{0} \end{aligned} \tag{1}$$

Some aesthetic qualities of the solutions can be very difficult to formalize and most consumer preferences are plainly subjective. It is therefore necessary to let them express their preferences during the optimization process. This approach, called interactive multiobjective optimization, directs the optimization process toward solutions that satisfy the user. Typically, the user must be able to choose from a set of optimized solutions and possibly to relaunch the optimization system if not satisfied. An interactive multiobjective optimization presents several advantages: The user can "learn about the interdependencies in the problem as well as about one's own preferences" (p. 3 in Ref. [28]), this can save some computational costs as the user's indication directs efficiently the search, and it avoids the need to compare many Pareto optimal solutions simultaneously (see p. 28 in Ref. [29]).

Since the investigated structures derived from mathematics or nature are highly nonlinear and often discrete, as can be the constraints and objectives, a stochastic solver instead of a classical (e.g., gradient-based) optimization approach is to be favored. A type of algorithm suitable for these types of problems is the genetic algorithm (GA) [30]. A GA tries to artificially simulate the process of evolution [31], by which the structures in nature were first created. For a review of applications using GA for multiobjective optimization, see Ref. [32]. The interactive genetic algorithm (IGA) approach is used when the user intervenes during the optimization process. The purpose of the user interaction is slightly different from that of classical IGA approaches, see, e.g., Ref. [29]. The user does not help the system to minimize $\mathbf{F}(\mathbf{x})$ by bringing in expertise (the results presented to the user are already optimized) but chooses the alternatives that are thought to fit the preferences and restarts the optimization if not satisfied.

Because some constraints can require long computational times (e.g., finite element analyses (FEAs)), it is preferable to handle the constraints sequentially, as this does not require that all constraints be evaluated at each iteration. This is implemented by scoring each individual according to which constraints it has fulfilled (see Sec. 4.4.2). The scoring system ensures that an individual that passes the m first constraints will always have a higher score than an individual that passes the $m-k$ first constraints ($k=1,\ldots,m-1$). Gradually, the number of individuals fulfilling the constraints will increase. Individuals that fulfill all constraints are scored based on how well they minimize $\mathbf{F}(\mathbf{x})$; this score is obtained by computing a weighted sum of the different objectives. The individuals are then ranked according to their scores. This ranking is used as basis for the selection of individuals with which to create the next generation of new individuals. The scoring and selection algorithm for the application is developed in Sec. 4.4. The ranking system for our application is presented in Sec. 4.4.2. This way of handling constraints sequentially, or lexicographically, will be denominated, hereafter, lexicographic constraint-handling technique (Lexcoht).

In summary, in order to solve the class of problems defined above, our approach is to model them as interactive multiobjective optimization problems, using stochastic solvers such as GA. In line with this approach, the following process is proposed.

The first step is, naturally, to let the designer or consumer customize parts of the object. Some of these will be required for the initialization of the optimization process. This can be material choice, contour, and choice of morphology. In the application presented below, the user first specifies the tabletop outline and the legs; the Voronoi structure was imposed. The optimization system is then launched and follows the same steps as a classical evolutionary optimization system. Once a first optimization run has been performed, a set of optimized products fulfilling all constraints is proposed to the user. If the user finds one that is deemed preferential, the process stops. Otherwise, the user is asked to choose a set of alternatives that are nearest the preferences, and the optimization process is rerun with the chosen individuals as "parents" for the initial population. The user can perform several iterations until completely satisfied, or give up.

4.2 Application

4.2.1 The Type of Structure. Among the infinite number of possible mathematical structures, cf. Ref. [1], the choice went to utilize a simple 2D tessellation, namely, the Voronoi diagram, which is easy to generate programmatically. Phenomena as diverse as the wing of a dragonfly, the structure of bone marrow, and a honeycomb can be described with Voronoi diagrams [33]. Such structures are often found in lightweight and strong structures in nature [2,34].

The Voronoi diagram is created from a number of Voronoi sites, or points. Each Voronoi site s is contained in a Voronoi cell, which contains all points closer to s than to any other Voronoi site. In the case of two sites s_1 and s_2, the space is divided in two by a straight line, which is the bisecting line of the segment $[s_1 s_2]$. If all sites are coplanar, the *Voronoi* diagram consists of polygons, see Fig. 1(*a*). Formally, a Voronoi diagram is described as follows. Let S be a set of n sites in Euclidean space of dimension d. For each site p of S, the Voronoi cell $V(p)$ of p is the set of points that are closer to p than to other sites of S. The Voronoi diagram $V(S)$ is the space partition induced by Voronoi cells (Ref. [35], Chap. 7).

4.2.2 The Product. Furniture has always been a suitable canvas for designers on which to paint future technologies and aesthetics [36]. In this sense, furniture provides a suitable test bed for innovation because any person understands what furniture represents. The furniture industry is also an important part of Swedish industry, representing 25 billion SEK (3.1 billion USD as of September 2009) [37]. Furniture is important also in terms of image in relation to what is widely known as Scandinavian design.

Moreover, tables pose high demand on low weight, stiffness, and visual appeal. This makes them suitable as illustration objects. Three tables with different sizes and loads have been chosen for this test: a coffee table, a side table, and a dining table.

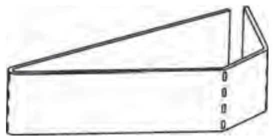


Fig. 2 Representation of a laser cut and bent Voronoi cell

The Voronoi diagram is used as the supporting structure of the table. The tabletop itself is made of glass so that the Voronoi structure is visible.

4.2.3 The Manufacturing Method. The following numbers of manufacturing methods were considered:

1. laser cutting strips of sheet metal and robot-welding them together
2. laser cutting strips of sheet metal and computer numerical control (CNC)-bending them into individual Voronoi cells and assembling them by welding, screwing, or gluing
3. same as method 2 but with laser cut perforations along the edges of the cells
4. sheet metal corrugated along the cell walls of the Voronoi diagram

Method 2 could allow the table to be assembled by the customer and the cells are easily individualized with different materials and colors. This is not possible with method 4 and very difficult with method 1 (because of welding issues). Method 3 allows for the same freedom as method 2 and furthermore makes it possible for the table to be bent by the customers (see Fig. 2). This reduces manufacturing and transportation costs (as the cells can be flat-packed) while the customers have the option to purchase a cheaper table (at the expense of added assembly time). In front of those arguments, both manufacturing methods 2 and 3 were chosen in order to let each customer decide of the assembly type she or he prefers.

4.2.4 Possibilities for Customization. For this product, the user/customer is able to fully define the contour and dimensions of the tabletop (see Fig. 3(*a*)). The user also chooses the table height, and the number and position of the legs. For each user-defined leg position, the Voronoi cell closest to it is set as a leg. The legs themselves are formed by the walls of cells, and have the same height as the table. They are fixed in all degrees of freedom during the finite element evaluation.

At the end of the first optimization run, the customer chooses the table that maximizes her or his preferences. The tables subsequently displayed have all fulfilled the constraints and have different costs (that is the cost minimization function presenting different values). Typically, four or five out of a population of 50 are presented to the user. The initial population for the next optimization run is then created by mutating and crossing the individuals selected by the user. The algorithm goes on until the user is satisfied (in this example, only two runs were performed).

4.3 Specifications. Beyond the table properties specified by the user/customer (Sec. 4.2.4), the inherent constraints necessary for a table to perform its function, the manufacturing requirements and cost (the manufacturing cost of the table must be minimized), were taken into account.

4.3.1 Functional and Aesthetic Constraints. The tables need to be able to handle the weight put on them without buckling, exceeding the yield stress of the material or deforming noticeably. A load of 500 N was used for the coffee table and the side table, and a load of 1000 N was used for the dining table to model the typical vertical loads. It is also important that the deformation of the table structure is not noticeable to someone sitting at the table,

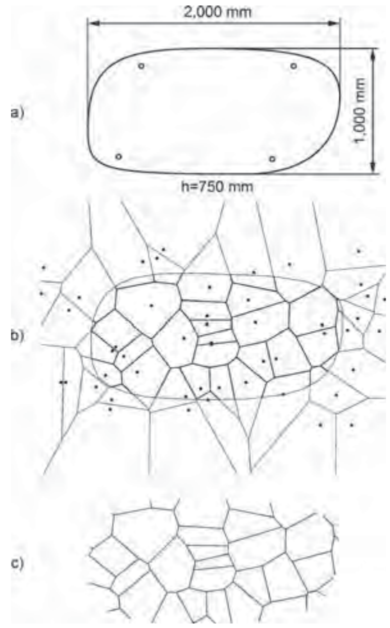


Fig. 3 (*a*) User-defined outline and leg positions of the table (dining table). (*b*) Example of how the Voronoi structure is created. The Voronoi diagram is generated from the Voronoi sites and cut off at the table boundary. (*c*) Final appearance of the structure after the cells have been cut off.

as the tabletop is in transparent glass. It was decided arbitrarily that a deformation under $a_{\mathrm{allowed}}=2.5$ mm would not be perceptible. This constraint is expressed as

$$\max(a) \le a_{\mathrm{allowed}} \tag{2}$$

where $\max(a)$ is the largest displacement measured in an individual.

4.3.2 Manufacturing Requirements. All tables were optimized with respect to the requirements for bending with a CNC-bending machine (manufacturing method 2).

As the CNC-bending machine will intersect with itself if the distance between two bends is too short, the cells cannot have walls less than 30 mm (l_{allowed}), or the bends have sharper angles than 33 deg ($\alpha_{\mathrm{allowed}}$) [38]. This is expressed by the following constraints:

$$\min(l) \ge l_{\mathrm{allowed}} \tag{3}$$

where $\min(l)$ is the shortest cell wall found in an individual, and

$$\min(\alpha) \ge \alpha_{\mathrm{allowed}} \tag{4}$$

where $\min(\alpha)$ is the smallest angle measured in an individual.

4.3.3 Cost Objective. The number of cells to be produced mostly governs the manufacturing cost, rather than the amount of material used. Also, the time of assembly is mostly dependent on the number of cells [39]. This means that to minimize the cost of the product, the optimization should focus on the number of cells, rather than on the amount of material used. The cost is the only objective to optimize for this product. The goal is thus to *minimize* n, where n is the number of Voronoi cells in the table structure.

In summary, the table generation system has to

$$
\begin{aligned}
&\text{minimize } n\\
&\text{subject to } \max(a) \leq a_{\text{allowed}}\\
&\min(l) \geq l_{\text{allowed}}\\
&\min(\alpha) \geq \alpha_{\text{allowed}}
\end{aligned}
\tag{5}
$$

4.4 Implementation

4.4.1 Representation of the Voronoi Diagram. To give the optimization algorithm full control of the Voronoi structure, the structures were represented by the location of the Voronoi sites. Each table was represented by the coordinates of $n_V=70$ Voronoi sites. The number of Voronoi sites was determined by experiment ranging from 50 to 100. With this range, the GA had enough points to construct a valid table structure while not having too many points to optimize. The number of sites could not be varied during the search. Nevertheless, the sites can move across the table boundary (Fig. 3(*b*)) to reduce or increase the number of cells in the table structure.

$$
\text{genome} = \begin{pmatrix} x_{1,1} & x_{2,1} \\ x_{1,2} & x_{2,2} \\ \vdots & \vdots \\ x_{1,70} & x_{2,70} \end{pmatrix}
\tag{6}
$$

To mutate the structures, the Voronoi sites were randomly moved by a quantity $\delta_{i,j}$, varying between 0% and 10% around $x_{i,j}$ at the beginning and decreasing linearly until the maximum number of generations is reached. To crossover two individual solutions, their vector of Voronoi site coordinates was exchanged at a random mutation point p.

4.4.2 Fitness Function and Selection. The constraints were handled, following the Lexcoht method described in Sec. 4.1. At each generation and for each individual, the structural requirements were tested first, then the cell wall length, and, finally, the cell angle. The individuals that possessed a larger displacement than allowed ($a_{\text{allowed}}=2.5$ mm) were scored the lowest. The score is based on Eq. (7*a*) with $\max(a)$ being the largest displacement a measured in the individual. The individuals passing the structural requirement but not the shortest wall requirement ($l_{\text{allowed}}=30$ mm) were scored the second lowest. The score is computed using Eq. (7*b*), where $\min(l)$ is the shortest cell wall found in an individual. Individuals passing the first two requirements but containing cells with bending angles sharper than $\alpha_{\text{allowed}}=33$ deg got a score according to Eq. (7*c*), where $\min(\alpha)$ is the smallest angle measured in an individual. Finally, the individuals that passed all the constraints were given a score p_4 inversely proportional to their number of cells, Eq. (7*d*), where n is the number of Voronoi cells in the structure.

$$
f_1 = 0 + \begin{cases} \dfrac{a_{\text{allowed}}}{\max(a)} & \text{if } \max(a) \geq a_{\text{allowed}} \\ 1 & \text{else(constraint 1 fulfilled)} \end{cases}
\tag{7a}
$$

$$
f_2 = 1 + \begin{cases} \dfrac{\min(l)}{l_{\text{allowed}}} & \text{if } \min(l) \leq l_{\text{allowed}} \\ 1 & \text{else(constraint 2 fulfilled)} \end{cases}
\tag{7b}
$$

$$
f_3 = 2 + \begin{cases} \dfrac{\min(\alpha)}{\alpha_{\text{allowed}}} & \text{if } \min(\alpha) \leq \alpha_{\text{allowed}} \\ 1 & \text{else(constraint 3 fulfilled)} \end{cases}
\tag{7c}
$$

$$
f_4 = 3 + \frac{n_V - n}{n_V - 1}
\tag{7d}
$$

An individual that does not fulfill any constraint gets the score f_1 and the other scores are discarded. Likewise, an individual that passes the first constraint but not the second gets the score f_2 and an individual that passes the first constraint but not the second gets the score f_3. Finally, an individual that passes all constraints gets the score f_4. The scores f_1, f_2, f_3, and f_4 have been normalized and offset such that

$$
0 < f_1 \leq 1 < f_2 \leq 2 < f_3 \leq 3 < f_4 \leq 4
\tag{8}
$$

With Eq. (8), an individual fulfilling m constraints is certain to get a higher score f_m than an individual fulfilling $m-k$ constraints (and gets a score f_{m-k}, $k=1,\ldots,m-1$). The advantage of this scoring system over a weighted sum is that if some constraints are not feasible, this can be discovered quickly.

After the scoring is done, a number of individuals are selected to populate the next generation. The individuals are ranked, following their score. The probability for an individual solution to be selected is related to its ranking.

The scoring and selection algorithm is represented in Fig. 4.

4.4.3 Termination. To ensure the convergence of the optimization, while still keeping the feedback time to the user reasonable, the termination of the optimization was set to occur after 600 generations, which is a moderate number of generations for truss optimization problems according to Giger and Ermanni [40].

4.5 The Table Generation System Software. The code was developed in MATLAB®. An interface has been developed for the user to draw the table boundary and indicate the leg positions. The implementation of GA is based on MATLAB's® genetic algorithm solver in the global optimization toolbox. To evaluate the structural stability, a finite element package developed at Lund University (Lund, Sweden) denoted CALFEM® [41] was used. This package makes it possible to calculate displacements and stresses in a structure by defining the structure's degrees of freedom, their coordinates, how they are connected to each other, and the boundary conditions for each degree of freedom. Each cell wall was represented by a beam element. The load was applied evenly across the nodes of the frame to simulate an even pressure from the glass top. The nodes that were meant to be leg nodes and lead down to the floor were set as fixed in all six degrees of freedom. Using the functions provided by CALFEM®, it was then possible to analyze displacements and stresses in the structure. MATLAB® has its own function to generate a Voronoi diagram [42].

4.6 Results. Three different boundaries were used to test the application: a dinner table of dimensions 2000 mm(L) × 1000 mm(B) × 750 mm(H), see Fig. 3(*a*), a coffee table (1000 × 1000 × 250 mm^3), and a side table (500 × 500 × 500 mm^3). Two runs were performed before the final tables were chosen. The first search for suitable individuals used a population of 50 individuals and 600 generations (a moderate number of generations for truss problems, as pointed out in Ref. [40]). The search took approximately 1.5 h of CPU time on a single core 3.0 GHz processor. After the first search was done, the user was presented with the different possible solutions. The selected individuals were then further optimized for another 600 generations in separate searches; the resulting best individuals from the different populations were then presented to the user for a final choice. An example of the result for the coffee table is presented in Fig. 5.

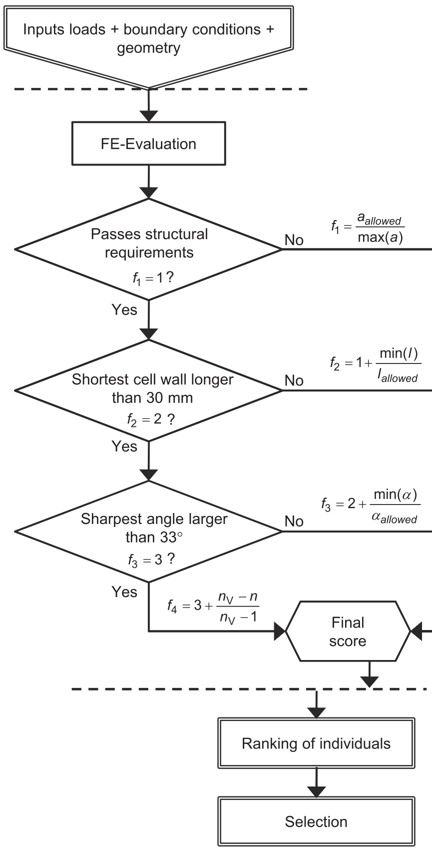


Fig. 4 Diagram of the evaluation function

Fig. 5 The final optimized structure of the coffee table

The resulting table structures were studied in detail in ANSYS® that confirmed that the structural constraints were respected. Prototypes of all three tables have been built and were exhibited at international design fairs.

5 Discussion and Conclusion

This paper has proposed an approach that addresses three issues relevant to the industry: (1) the low integration of industrial design into the product development process, (2) the limited morphological repertoire currently available to designers, and (3) the limited possibilities of product form customization. These issues have been tackled separately in literature but have not been integrated. The proposed approach is to couple complex morphologies with an interactive optimization system. By using complex morphologies, designers can deal with forms they could scarcely imagine. Systems containing these formalized structures can then be optimized, taking into account aesthetical and functional engineering and manufacturing constraints. The user/customer can customize parts of the object, some of which will be required for the initialization of the optimization process (material, contours, and morphology), and then interact with the optimization system by selecting the resulting products according to her or his preferences. This approach allows for a true mass-customization without resorting to rapid prototyping. With an integration of morphologies with engineering and manufacturing constraints, the iterations between industrial designers and engineers are reduced.

The optimization problem contains a mixed (continuous and discrete) set of constraints, and the morphologies are not described by linear equations; therefore, stochastic search algorithms, such as GA, are recommended for finding solutions. Special emphasis must also be put on constraints: The solution space for engineering problems is often small and scarce, and some constraints are either hard or time-consuming to fulfill (requires extensive FEA). In our application, they have been prioritized accordingly.

In our approach, we propose that the multi-objective minimization be handled by a weighted sum. With the Lexcoht approach, the individuals fulfilling all the constraints can readily minimize the objective functions while the others are still evolving in other areas of the search space. However, the weighted sum is not always an efficient multi-objective optimization approach and the setting of the weights may be an arduous task (p. vi in Ref. [43]). Alternatively, one could separate the constraint-handling activity from the objective optimization (see, e.g., Ref. [44]), allowing the use of many more multi-objective optimization techniques. Making the population evolve until a certain percentage of the population fulfills all constraints could do this. These individuals would then be used for the optimization of the objective functions. This promising alternative requires further research.

We have now successfully tested our approach with another type of furniture (a bookshelf, see Ref. [45]). It is not difficult to imagine many other applications for the use of morphologies from nature and mathematics in design. For example, many 2.5D objects can be designed with regular or irregular tessellations (see Fig. 1(*c*)), furniture (as above), flooring, and wall elements are obvious examples, others are façade elements (window grates and balustrades), enclosure elements (wind deflectors and noise barriers), driveway elements (drainage gates and banisters), etc. (a product typology of such objects is available upon request). Our approach may also be used for parts of other products as long as the interface with the other parts is well defined. It could lead to new business strategies and models, building on augmented design automation and customer involvement instead of the traditional business model development-manufacturing-distribution-consumption.

The table generating system takes some engineering and manufacturing constraints into account but the potential issues linked with automating the production preparation (process planning and computer-aided manufacturing (CAM)) and detailed FEA need to be further investigated. The use of more advanced algorithms may enhance the heuristic proposed.

The table generating system in its current form does not allow for instant feedback. Even efficient, fully parallelized algorithms, while reducing the search time dramatically, will not solve this general problem. Generative systems may simply take too long and, if the system is not designed carefully, the user may get to a point, where no design can be generated, given the constraints imposed. The impact of these issues on the user is under investigation, see Ref. [45]. It is necessary to consider usability features in such systems with special attention.

Another important point is the effort required to develop a dedicated application. Even if a high degree of freedom is conceded to the user, many form characteristics have to be frozen during the application development. In our example, the tables were optimized with constant Voronoi cell height. Nevertheless, the tables may be considered visually more pleasing and interesting if the cell heights in the final models are varied.

Acknowledgment

The authors would like to thank Associate Professor Giorgos Nikoleris at the Division of Machine Design, Lund University for his expert advice on the characteristics of the different manufacturing systems. We also thank the reviewers for their valuable comments regarding our approach.

References

[1] Hopf, A., 2009, "Renaissance 2.0—Expanding the Morphologic Repertoire in Design," 24th Cumulus Conference 24/09, pp. 78–85.
[2] Pearce, P., 1978, *Structure in Nature is a Strategy for Design*, MIT, Cambridge, MA.
[3] Oxman, R., 2006, "Theory and Design in the First Digital Age," Des. Stud., **27**(3), pp. 229–265.
[4] Shea, K., 2004, "Explorations in Using an Aperiodic Spatial Tiling as a Design Generator," First Design Computing and Cognition Conference DCC '04, pp. 137–156.
[5] Vincent, J. F. V., and Mann, D. L., 2002, "Systematic Technology Transfer From Biology to Engineering," Philos. Trans. R. Soc. London, Ser. A, **360**(1791), pp. 159–173.
[6] Chakrabarti, A., Sarkar, P., Leelavathamma, B., and Nataraju, B. S., 2005, "A Functional Representation for Aiding Biomimetic and Artificial Inspiration of New Ideas," Artif. Intell. Eng. Des. Anal. Manuf., **19**(2), pp. 113–132.
[7] Velcro, S. A., 1955, "Improvements in or Relating to a Method and a Device for Producing a Velvet Type Fabric," Swiss Patent No. 721338.
[8] Hutchinson, H., 2008, "Beyond the Shark Skin Suit," Mech. Eng. (Am. Soc. Mech. Eng.), **130**(4), pp. 42–44.
[9] McCormack, J., Dorin, A., and Innocent, T., 2004, "Generative Design: A Paradigm for Design Research," Futureground.
[10] Knight, T. W., 1980, "The Generation of Hepplewhite-Style Chair-Back Designs," Environ. Plann. B, **7**(2), pp. 227–238.
[11] Orsborn, S., Cagan, J., Pawlicki, R., and Smith, R. C., 2006, "Creating Cross-Over Vehicles: Defining and Combining Vehicle Classes Using Shape Grammars," Artif. Intell. Eng. Des. Anal. Manuf., **20**(3), pp. 217–246.
[12] McCormack, J. P., Cagan, J., and Vogel, C. M., 2004, "Speaking the Buick Language: Capturing, Understanding, and Exploring Brand Identity With Shape Grammars," Des. Stud., **25**(1), pp. 1–29.
[13] Pugliese, M. J., and Cagan, J., 2002, "Capturing a Rebel: Modeling the Harley–Davidson Brand Through a Motorcycle Shape Grammar," Res. Eng. Des., **13**(3), pp. 139–156.
[14] Chau, H. H., Chen, X., McKay, A., and de Pennington, A., 2004, "Evaluation of a 3D Shape Grammar Implementation," First Design Computing and Cognition Conference DCC '04, pp. 357–376.
[15] Shea, K., and Cagan, J., 1999, "Languages and Semantics of Grammatical Discrete Structures," Artif. Intell. Eng. Des. Anal. Manuf., **13**(4), pp. 241–251.
[16] Trubidge, D., 2004, "Coral [Lamp]," *Desire—The Shape of Things to Come*, R. Klanten, S. Ehmann, A. Kupetz, S. Moreno, and A. Mollard, eds., Die Gestalten Verlag, Berlin, p. 113.
[17] Wertel, J., and Oberfell, G., 2007, "Fractal-T [Table]," *Desire—The Shape of Things to Come*, R. Klanten, S. Ehmann, A. Kupetz, S. Moreno, and A. Mollard, eds., Die Gestalten Verlag, Berlin, p. 134.
[18] van der Veer, B., 2007, "Paper [Table]," *Desire—The Shape of Things to Come*, R. Klanten, S. Ehmann, A. Kupetz, S. Moreno, and A. Mollard, eds., Die Gestalten Verlag, Berlin, p. 99.
[19] Mayor, J., 2007, "Burnout Bench [Bench]," *Desire—The Shape of Things to Come*, R. Klanten, S. Ehmann, A. Kupetz, S. Moreno, and A. Mollard, eds., Die Gestalten Verlag, Berlin, p. 117.
[20] Sharfman, B., 2006, "Mercedes and the Boxfish," The Scientist, **20**(9), pp. 17–18.
[21] Cheng, K., Harrison, D. K., and Jiang, Z., 2002, "An Integrated Concurrent Engineering Approach to the Design and Manufacture of Complex Components," Int. J. Adv. Manuf. Technol., **20**(5), pp. 319–325.
[22] Simpson, T. W., Siddique, Z., and Jiao, J. R., 2005, *Product Platform and Product Family Design: Methods and Applications*, Springer, New York.
[23] Huang, G. Q., Li, L., and Chen, X., 2007, "A Tandem Evolutionary Algorithm for Platform Product Customization," ASME J. Comput. Inf. Sci. Eng., **7**(2), pp. 151–159.
[24] Lakhani, K. R., and Kanji, Z., 2009, Threadless: The Business of Community, Harvard Business School Multimedia/Video Case 608–707, Harvard Business School, Cambridge, MA.
[25] von Hippel, E., 1988, *The Sources of Innovation*, Oxford University Press, New York, NY.
[26] von Hippel, E., 2005, *Democratizing Innovation*, MIT, Cambridge, MA.
[27] Blecker, T., and Friedrich, G., 2006, *Mass Customization: Challenges and Solutions*, Springer Science+Business Media, Inc., Boston, MA.
[28] Miettinen, K., 2008, "Introduction to Multiobjective Optimization: Noninteractive Approaches," *Multiobjective Optimization: Interactive and Evolutionary Approaches*, D. Hutchison, J. Branke, K. Deb, T. Kanade, J. Kittler, J. M. Kleinberg, F. Mattern, K. Miettinen, J. C. Mitchell, M. Naor, O. Nierstrasz, C. Pandu Rangan, R. Slowinski, B. Steffen, M. Sudan, D. Terzopoulos, D. Tygar, M. Y. Vardi, and G. Weikum, eds., Springer, Berlin, pp. 1–26.
[29] Miettinen, K., Ruiz, F., and Wierzbicki, A. P., 2008, "Introduction to Multiobjective Optimization: Interactive Approaches," *Multiobjective Optimization: Interactive and Evolutionary Approaches*, D. Hutchison, J. Branke, K. Deb, T. Kanade, J. Kittler, J. M. Kleinberg, F. Mattern, K. Miettinen, J. C. Mitchell, M. Naor, O. Nierstrasz, C. Pandu Rangan, R. Slowinski, B. Steffen, M. Sudan, D. Terzopoulos, D. Tygar, M. Y. Vardi, and G. Weikum, eds., Springer, Berlin, pp. 27–57.
[30] Koza, J. R., Keane, M. A., Streeter, M. J., Adams, T. P., and Jones, L. W., 2004, "Invention and Creativity in Automated Design by Means of Genetic Programming," Artif. Intell. Eng. Des. Anal. Manuf., **18**(3), pp. 245–269.
[31] Holland, J. H., 1975, *Adaptation in Natural and Artificial Systems: An Introductory Analysis With Applications to Biology, Control and Artificial Intelligence*, University of Michigan Press, Ann Arbor, MI.
[32] Coello, C. A., 2000, "An Updated Survey of GA-Based Multiobjective Optimization Techniques," ACM Comput. Surv., **32**(2), pp. 109–143.
[33] Aurenhammer, F., 1991, "Voronoi Diagrams—A Survey of a Fundamental Geometric Data Structure," ACM Comput. Surv., **23**(3), pp. 345–405.
[34] Beukers, A., and van Hinte, E., 2005, *Lightness: The Inevitable Renaissance of Minimum Energy Structures*, 4th ed., 010 Publishers, Rotterdam.
[35] de Berg, M., Cheong, O., van Kreveld, M., and Overmars, M., 2008, *Computational Geometry—Algorithms and Applications*, Springer-Verlag, Berlin.
[36] Fayolle, C., 2006, "The Future is Less to Discover Than to Invent (in French. Original Title: L'avenir est moins à découvrir qu'à inventer)," Biennale Internationale Design Saint-Etienne, Cité du Design, Saint-Etienne, E. Francès, ed., Vol. 336, p. 126.
[37] TMF/SCB, 2009, TMF in Figures—Statistics on the Swedish Tree and Furniture Industry (in Swedish. Original Title: TMF i siffror—Statistik om den svenska trä—och möbelindustrin), No. 1, MTF, Stockholm.
[38] FERRUM Lasercut GmbH, personal communication.
[39] Linde Metall AB, personal communication.
[40] Giger, M., and Ermanni, P., 2006, "Evolutionary Truss Topology Optimization Using a Graph-Based Parameterization Concept," Struct. Multidiscip. Optim., **32**(4), pp. 313–326.
[41] Austrell, P. E., Dahlblom, O., Lindemann, J., Olsson, A., Olsson, K.-G., Persson, K., Petersson, H., Ristinmaa, M., Sandberg, G., and Wernberg, P.-A., 2004, *CALFEM: A Finite Element Toolbox—Version 3.4*, Structural Mechanics LTH, Lund University, Lund.
[42] Mathworks, 2009, "Matlab Documentation—Voronoin Function," ⟨http://www.mathworks.com/access/helpdesk/help/techdoc/index.html?/access/helpdesk/help/techdoc/ref/voronoin.html⟩, last accessed: Sep. 22.
[43] Hutchison, D., Branke, J., Deb, K., Kanade, T., Kittler, J., Kleinberg, J. M., Mattern, F., Miettinen, K., Mitchell, J. C., Naor, M., Nierstrasz, O., Pandu Rangan, C., Slowinski, R., Steffen, B., Sudan, M., Terzopoulos, D., Tygar, D., Vardi, M. Y., and Weikum, G., 2008, *Multiobjective Optimization: Interactive and Evolutionary Approaches*, Springer, Berlin.
[44] Schoenauer, M., and Xanthakis, S., 1993, "Constrained GA Optimization," Fifth International Conference on Genetic Algorithms—ICGA '93, pp. 573–580.
[45] Nordin, A., Motte, D., Hopf, A., Bjärnemo, R., and Eckhardt, C.-C., 2010, "Complex Product Form Generation in Industrial Design: A Bookshelf Based on Voronoi Diagrams," Fourth Design Computing and Cognition Conference—DCC '10, pp. 701–720.

Paper III

III

Complex product form generation in industrial design: A bookshelf based on Voronoi diagrams

Nordin, A., Motte, D., Hopf, A., Bjärnemo, R., & Eckhardt, C.-C.

4th Design Computing and Cognition Conference - DCC'10,
Stuttgart, July 12-14, 2010, pp. 701-720.

Originally published by Springer

Complex Product Form Generation in Industrial Design: A Bookshelf Based on Voronoi Diagrams

Axel Nordin, Damien Motte, Andreas Hopf, Robert Bjärnemo and Claus-Christian Eckhardt
Lund University, Sweden

Complex product form generation methods have rarely been used within the field of industrial design. The difficulty in their use is mainly linked to constraints – such as functionality, production and cost – that apply to most products. By coupling a mathematically described morphology to an optimisation system, it may be possible to generate a complex product form, compliant with engineering and production constraints. In this paper we apply this general approach to the designing of a bookshelf whose structure is based on Voronoi diagrams. The algorithm behind the developed application used here is based on a prior work submitted elsewhere [1], adapted to the bookshelf problem. This second example of product form generation, which includes specific constraints, confirms the relevance of the general approach.

The handling of complex morphologies is not straightforward. Consequently, an explorative study on that theme has been performed. A user interface has been developed that allows for designing a bookshelf based on Voronoi diagrams. The user interface was subsequently tested by peer designers. The results suggest that user attitudes diverge: one faction preferred maximum freedom of creation, that is, maximum control of the form creation process; the other faction wanted the application to generate a bookshelf based on their functional needs (e.g. adapt to the number and types of objects to be stored) and would ask for a "surprise me" effect for the final solution.

Design Computing and Cognition DCC'10. J.S Gero (ed), pp. 691-710.

Introduction

Although complex – mathematical or nature-inspired – form generation methods have long been employed in the field of architecture [2, p. 137], this has rarely been the case in industrial design. One barrier for such development in the latter discipline is the multitude of constraints linked to the form-giving of products; surfaces are often functional, the artefacts are produced in several exemplars – meaning that the product form must be modified to suit production systems; cost control is consequently important; finally, engineering constraints must also be respected. Another obstacle may be the lack of educational initiation in industrial design.

However, the situation is beginning to evolve; the ongoing digitalisation of the entire product design activity simplifies access to form generation tools whilst digital fabrication facilitates the production of physical prototypes. This digitalisation should allow for a much tighter integration of industrial design, engineering and production. Last but not least, one can sense an evolution of the by and large static relationship between the consumer and the product. There is an increasing desire to participate in the designing of products and the potential experiences consumers will share with them. As put forward by Friebe and Ramge [3], the upsurge of independent fashion labels, *crowdsourcing* initiatives or co-working spaces indicates the demand for consumer *empowerment*. This need for co-creation, implemented already in textile [4] but also in more advanced consumer goods businesses like sportswear, [5] and [6], goes well beyond mere material and colour choice – the future *prosumer* [7] desires control over form and function as well. Generative design can be one facilitator in such developments.

In a prior work [1], we have begun to study the use of complex forms in industrial design, taking into account functional, engineering and production constraints. The term complex is to be understood here in the sense that the form is virtually impossible to generate without computer aid. The present publication has two goals. First, we aim at partially validating the approach proposed in [1] by investigating another product type with different kinds of constraints and objectives. Second, the handling of complex forms is not straightforward. Most users (designers or consumers) cannot or do not want to manipulate directly the parameters linked to a morphology; in some cases this is even impossible. It is necessary to find alternative ways of controlling form that make sense for the user. We consequently reflect on the way the user can interact with these complex forms, and a user interface allowing for the development of the bookshelf based on Voronoi diagrams has been developed and tested.

Background

Expanding the morphologic repertoire in design

The morphologic repertoire is the infinite repository of all two- and three-dimensional forms, structures and compositions thereof. Although no morphology has *a priori significance* – its adequacy pertaining only to the intended usage criteria – a designer's command of an extensive morphologic vocabulary and grammar enhances creative expressivity, which, in turn, is no end in itself, but essential for a designer's ability to rise to the present and future economic and ecologic challenges [8].

The prevalent *modus operandi* concerning the form-giving activity in industrial design is characterised by explorations that depend on the individual capability to mentally manipulate a solution space from which to select and express the intended result. In that sense a designer or team of designers is equivalent to an *auteur*, because the initial objective and resultant object are inextricably linked by a volitional act [9]. The morphologic repertoire, on which the form-giving activity is based, is by and large rooted in artistic experimentation (serendipity), cursory inspiration (mimicry) or canonical stipulation (methodology). Reliance on such rather traditional approaches is not problematic *per se*; a trained designer generally will come out with a *satisfying* solution to a brief.

However, individualist or formal aesthetic motivations preclude the creative potential of *generative* mathematical and natural morphologies that could be equally inspiring points of departure. Even more importantly, once these morphologies are coupled to algorithmic design processes, they provide access to *performative* and *emergent* qualities only found in dynamic systems [9]. Algorithmically controlled morphologies not only pave the way for the unimaginable, they also present methods to handle and adapt them to an intended purpose. In that sense, the form-giving activity is augmented or rather transmuted into a *form-finding* process – an almost literary *meta-design* activity concerned with the formulation of rules and constraints from which desired or unintended, but feasible, results emerge. Quite possibly, the self-conception of what a designer is and does will change considerably in the future; designers may eventually become *scriptwriters*, *moderators* or *curators* – or even redundant altogether?

Form generation in the larger context

Apart from providing new creativity-enhancement tools, it is important to integrate them into the design context. As mentioned above, an industrial designer’s activities in the product development process are intertwined with engineering and production preparation activities. Nevertheless, even in organisations were these different functions are well integrated, iterations are still unavoidable. Taking advantage of the digitalisation of engineering and production preparation activities, efforts have been made taking into account their different constraints early on in the development process (see e.g. [10]). In the latter context, the industrial design activity is still somewhat overlooked. By integrating critical engineering and production requirements in the design process, the likelihood for a designer to “get it right first time” – or at least to reduce the number of critical changes in the design– is higher.

If form is algorithmically generated and engineering and production constraints integrated in the process, partial or full transfer of the design activity to consumers becomes a concrete option – new business models will emerge as a result. Many businesses have already implemented mass-customisation to some degree, with the automotive industry being a precursor. That approach has been adopted in other industries like in the case of Threadless® (clothing) [4] or Innovate with Kraft® (food) [11]. The demand for bespoke products and services is increasing, even in markets where branding is important. “The world has changed. Consumers interact with brands on their own terms,” says Trevor Edwards, Vice President, Brand and Category Management for Nike® [12]. Consequently, the NikeId® website and studios, [12] and [13], provide for high-level customisation of athletic footwear. Many consumers in saturated economies are decreasingly passive; it will be ever more important to hand over some control over the products-to-be they desire. Even if the traditional modes of product development remain dominant in the foreseeable future, the exploitation of niche markets becomes increasingly relevant and more profitable [14].

Handling complex forms requires the development of an adequate way for users to make sense of the creation process. The design environment is discussed in the section "User interaction".

Related works on generative product design systems

Generative design systems that take into account functional and technical constraints (engineering and production) as well as aesthetic intent have existed for long in the field of architecture while such systems have rather been the object of isolated research studies in industrial design.

In industrial design, generative design has primarily been used for stylistic purposes. In the seminal work of Knight [15], a parametric shape grammar was developed for the generation of Hepplewhite-style chairbacks. Orsborn et al. [16] employed a shape grammar to define the boundaries between different automotive vehicle typologies. Recent works have focused on branding related issues. With the help of shape grammars, new designs based on the Buick® [17], Harley-Davidson® [18], Coca-Cola® or Head & Shoulders® [19] brand were developed. Further research is undertaken towards rules that are linking form and brand (e.g. [20] for GA-based systems and [21] for shape grammars).

Some works are crossing the boundaries between engineering and industrial design, taking into account functional or technical constraints and aesthetics. Shea and Cagan [22] used a combination of shape grammar and simulated annealing for both functional and aesthetical purposes and applied it for truss structures (truss structures are commonly used for both heavy industrial applications and consumer products). Shape grammars are used to generate new designs while the simulated annealing technique directs the generation towards an optimum. The design objectives were functional (minimise weight, enclosure space and surface area), economic and aesthetic (minimise variations between lengths in order to get uniformity, make the proportions respect the golden ratio). Their model has been re-used in [23] (shape grammar and genetic algorithm, or GA) to develop stylistically consistent forms and has been applied to the design of a camera. The designs generated took into account the constraints linked to the spatial component configuration. A designer was in charge of the aesthetic evaluation, following the interactive genetic algorithm (IGA) paradigm. Ang et al. [23], also using shape grammars and GA, developed the Coca-Cola® bottle example of [19] and added functional considerations (the volume of the bottle), that were constrained to approach the classic Coca-Cola® bottle shape. Morel et al. [24], within the IGA paradigm, developed a set of chairs optimised for weight and stiffness. Finally, Wenli [25] developed a system that, through adaptive mechanisms, allows it to learn the designer’s intent faster; that system was implemented as a plug-in for a CAD system and applied to boat hull design.

Approach

In the works presented above, shape grammar is the main technique used. In our study, we use a pre-determined computational geometry (namely, Voronoi diagrams) instead and optimise the form according to engineering and production constraints. The use of mathematical and natural morphologies has not been the object of much applied research in industrial design, but has been implemented for the development of several products and prototypes in industry (see [1] for a coarse typology of such products). An important observation is that, in many cases, plastics rapid prototyping is the fabrication system of choice. Restricting the application of an extended morphologic repertoire to rapid prototyping may not be sustainable in the long term as only a few types of products are suitable for this fabrication technology. Rapid prototyping is likely not to be the panacea. That is why we focused on “traditional” production systems such as laser cutting and CNC sheet metal bending. Defining and evaluating the aesthetics is up to the user through the IGA paradigm. This continuation of Nordin et al. [1] aims first at partially validating that paradigm by using a different product (bookshelf vs. table), another material (phenolic film – PF – coated veneer core plywood), and consequently a different manufacturing system (circular saw and strip-grinder) and different constraints (addition of a functional constraint). Second, the focus is on how new forms can be practically handled. The user (designer or consumer) may not necessarily be interested in a certain mathematical or natural morphology *per se,* but rather in its aesthetic potential. It is quite difficult to handle complex morphologies. Proper controls have to be defined and, to that end, a dedicated interface has been created and tested.

Short on the Voronoi structure

A Voronoi (or Thiessen) structure is a simple 2D tessellation, as shown in **Fig. 1**. Structures based on a Voronoi structure (or Voronoi diagram) are often found in very robust yet lightweight structures in nature. Apart from aesthetic aspects, a Voronoi-based bookshelf would consequently have a structure well suited for carrying heavy loads such as books whilst maintaining a low weight. A Voronoi structure can be described as follows: Let $p_1,.., p_n$ be a set of n distinct points in the plane; these points are called the Voronoi sites. For each site, the set of points that are closer to it than to any other site form a Voronoi cell. A Voronoi diagram is constituted of all such cells. An overview of a Voronoi diagrams' properties can be found in [26, chapter 7].

The bookshelf

As in [1], the Voronoi structure is applied to a common type of furniture - a bookshelf. In the table case, the Voronoi structure was used as support for a glass tabletop, **Fig. 2**. In case of the bookshelf, every Voronoi cell serves as a storage compartment.

The manufacturing process consists of cutting, gluing and assembling PF coated veneer core plywood parts. Each Voronoi cell is manufactured as an individually cut and glued compartment. The critical constraints related to functionality and production methods are described later.

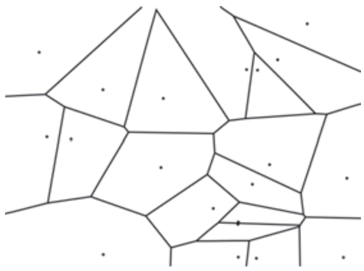


Fig. 1 Example of a Voronoi diagram

User interaction

The initial question regarding functionality, interactivity and output of a generative design and optimisation application is whether the designer or consumer is willing to relinquish control to a certain degree. An *auteur* designer-personality may hold the view that algorithms *seemingly* restrict creativity; an experimentally open-minded design-personality may, in contrast, actively seek for emergent behaviour to find unexpected solutions. . But it should be noted here that often, modern dance performances, music scores and contemporary architectures have been developed with help of algorithms – and the creativity of William Forsythe, Iannis Xenakis or Zaha Hadid has not been disputed. For consumers, control is important as it is not only the uniqueness of the product that matters, but also its personalisation [27].

However, in case the designer or consumer is attracted to generative design methodologies, the question then is in what way the degree of freedom is limited – and for what reason. The resultant output may either only remotely resemble the chosen start-up design, or turn out to be fairly predictable. One could therefore speak of *controlled serendipity*, wherein the number of constraints – whether aesthetic or functional – is the determining factor.

The sheer number of constraints – determining the degree of usability – must also be considered. Whereas a skilled designer may wish to *condition* a generative design and optimisation application with constraints beyond the aesthetic, e.g. complex functional and production constraints, an unskilled consumer is likely to not wanting to venture much beyond the aesthetic and overall dimensioning. Therefore, such application, if generalised for the widest possible range of input morphologies and output product typologies, needs a very customisable graphical user interface (GUI) in order to show and hide complexity depending on the task at hand. An interface in accordance with such conception of the design work with respect to an expanded morphology has been produced and is presented next. The acceptance of that interface has then been explored by letting a group of peer designers use it and elaborate on their experiences. Their feedback is presented in the second part of this section.

Fig. 2 Example of table developed in [1]. The Voronoi structure was used as support for a glass tabletop and was optimised with respect to deformation and CNC bending constraints.

The interface

In [1], users were able to define only the contour of a table and could not affect the layout of the Voronoi cells; in the case of the bookshelf it was decided to concede greater control to users – manipulating the compartmentalisation of the bookshelf for both aesthetic and functional purposes– and get feedback from a group of designers on that matter.

To enable users to control the appearance of the bookshelf, several strategies were discussed. It would have been possible to give them complete freedom to place the points of the Voronoi diagram. This, however, would have be a tedious task for someone not interested in controlling every aspect of the tessellation – and would be too complex for someone not familiar with the specific behaviour of Voronoi diagrams. Instead, it was decided to offer users the option of controlling the compartmentalisation via a set of parameters. The *staggering* and *randomness* parameters allow for the control of the generation of aesthetically interesting structures. *Staggering* controls the internal angles of the compartments so that the user can create cell-forms ranging from square to hexagonal. *Randomness* controls the randomisation of angles and compartment sizes.

Beside the functional parameters that determine the external dimensions of the bookshelf (*height*, *width* and *depth*), its usability is also very much dependent on compartment sizes and forms. To that end, it was decided to offer users three parameters ruling the distribution, size and form of the compartments, namely *growth*, *sparsity* and again *staggering* (the same parameter as above). *Growth* controls how the sizes of compartments are distributed to enable users to generate a bookshelf with small compartments on top and progressively larger compartments towards the floor. *Sparsity* controls the scale of the entire Voronoi diagram to allow users to uniformly scale the compartments up or down.

Additionally, three variables (*x-position*, *y-position* and *rotation*) enable the user to fine-tune the bookshelf structure. Users may want to avoid small compartments at the outer perimeter; rotating and/or shifting the entire structure horizontally and/or vertically can achieve this.

To reduce the complexity of handling nine variables, the creation process was divided into three steps – in the first step, the dimensions of the bookshelf are set; in the second step, its internal Voronoi structure is defined and in the third step, the structure can be fine-tuned.). The GUI is presented in **Fig. 3** with all steps shown at once. Finally, the bookshelf is optimised and presented to the user, **Fig. 4**.

Interface evaluation

To assess the application's usability and the selection of parameters, four professional industrial design peers tested the interface. They were asked to use the application to create and optimise one bookshelf per person and were questioned on the tool's usability and their opinion of the amount or lack of control over the creation process and final solution.

What became apparent from the usability testing was that the designers were split into two distinct factions – either wanting maximum freedom of creation, or wanting the application to generate bookshelves based on their needs.

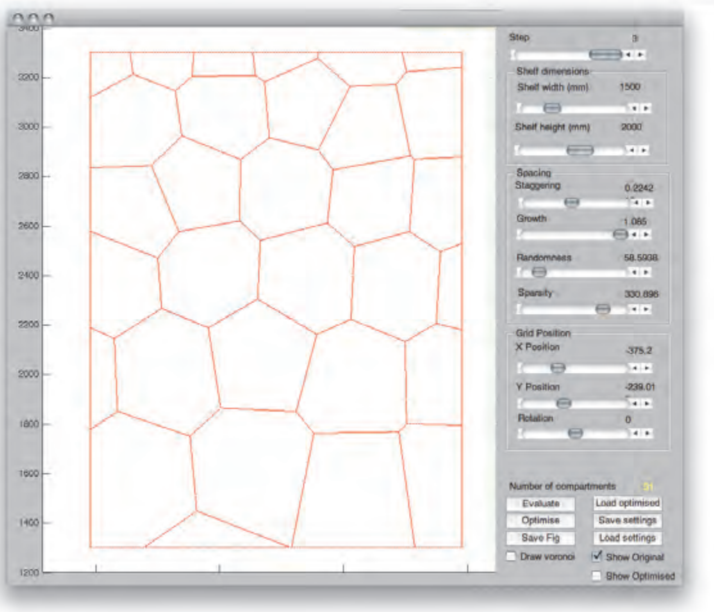


Fig. 3 The GUI for the bookshelf creation application (all steps shown).

The designers requiring maximum freedom wanted to be able to edit the Voronoi structure in detail by adding, removing and moving the Voronoi sites (points) as well as defining the bookshelf's contour. They felt deceived noticing that the algorithms had changed their structure after optimisation for production and they felt that their work had been taken away from them. They would have preferred to be able to get continuous visual feedback on the bookshelf' properties and adjust the structure themselves according to the feedback.

The other faction of designers requested a feature to input information on their particular requirements, such as the types of objects to be stored and/or the dimensions of a room – and then automatically receive a few optimised solutions fulfilling their needs to choose from.

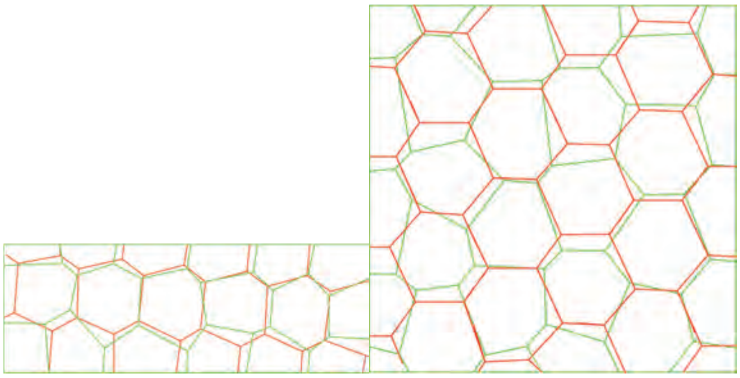


Fig. 4 The original user defined structure in black, optimised solutions in grey (in red, respectively green in the colour version)

The conclusion is that the application needs to accommodate both types of users; those requiring absolute control and those only interested in the bookshelf's functionality – something which could be achieved by either offering two modes of operation, or two entirely different applications. The goal of allowing users greater freedom in designing the structure was achieved, but evidently it will be necessary to concede to both types of users even more and/or different kinds of control.

Although at first feared to be a usability problem, the time users had to wait while the final solution was being optimised was not considered a major drawback in the application. In fact, some were willing to wait as long as one evening before getting the result – given that it would be satisfactory.

A photo-realistic rendering of the final solution was also requested – or at least an isometric representation including a depiction of the material thickness.

The general search algorithm

The algorithm for optimising the bookshelf's structure is essentially the same as in [1], with some adaptations for the present project. As described in the section above, the user "designs" the initial prototype that will be used to create the first generation of individuals. Only one final solution is presented to the user, who, if not satisfied, can re-launch the optimisation process an infinite number of times. The user can change the newly obtained bookshelf at each iteration step (subsequent testing by the designers showed that this was superfluous, see last section). Because the interaction is limited to choosing or not choosing the outcome of the optimisation, the principle of the algorithm is more of a “semi-interactive“ optimisation, than a usual IGA. The adapted algorithm is presented **Fig. 5**.

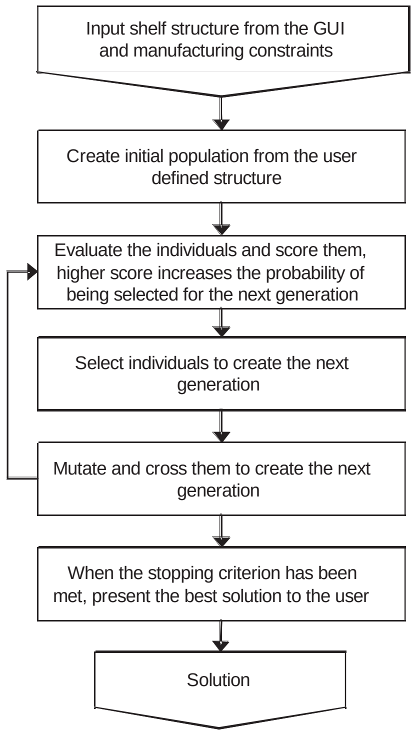


Fig. 5 Diagram of the general algorithm

The evaluation model for the multi-objective optimisation has also been modified to better suit the usability and production requirements of a bookshelf. In [1], a structural analysis was performed on the generated tables to ensure their stability – the most time-consuming evaluation step. Because a Voronoi structure is a robust tessellation and the material thickness can be over-dimensioned in the case of a bookshelf to eliminate the risk of structural failure, it was decided not to perform a structural analysis during the optimisation – but rather on the user's selected solution.

Simulations in [1] had also shown that one production constraint was more difficult to fulfil than others. These time-consuming tasks had resulted in a prioritisation of the resolution of the different constraints. In this case, such issues did not occur and the multi-objective optimisation model used is similar to [22], a weighted sum of both the constraints and the objective values.

The GA characteristics

As in [1], the *representation* of the structure to be optimised consists of the coordinates of the Voronoi sites, $p_1,.., p_n$. The number of Voronoi sites, n, depends on the user input.

$$genome = \begin{bmatrix} x_{1,1} & x_{2,1} \\ \vdots & \vdots \\ x_{1,n} & x_{2,n} \end{bmatrix} \quad (1)$$

The Voronoi diagram is created from the coordinates of these sites and cut off along the edges of the user-defined contour of the bookshelf (see e.g. **Fig. 3**). The resulting polygonal structure is used for evaluation (see below). The *mutation function* is identical to the one used in [1], and consists in displacing each coordinate, $x_{i,j}$ of the Voronoi sites in the representation by a random amount, $\delta_{i,j}$, varying between 0 and 10% of $x_{i,j}$. The maximum amount of change decreases linearly from the first generation to the last. The *crossover function* is also identical to the one used in [1], and consists in exchanging the coordinates of the Voronoi sites of two parents after a random mutation point to create two children.

As in [1] the *selection* is based upon ranking the individuals according to their fitness (the fitness function is described by Equation 5). An individual's probability for being selected is proportional to its ranking.

Duplicating the user-defined individual to fill the population creates the initial population; then the mutation function is applied to all individuals in the population apart from one, which remains unchanged.

The parameters of the search algorithm were chosen to give a feedback time of around one hour, which was deemed reasonable, as well as a high probability of receiving a solution that fulfilled all constraints. Therefore the stopping criterion was chosen to be the maximum number of generations, which was set to 300. The size of the population was set to 50 individuals. This proved to give feasible solutions to all user-defined structures during the testing. See Figure 7 for an example of a typical fitness curve.

Constraints, objectives and evaluation

For the table example of [1], it was required that the cells have walls longer than 30 mm and the internal angles are larger than 33° in order to be suited for production with a CNC sheet metal bending machine [28]. It was also necessary to take care of the stiffness and deformation (engineering constraint). In the case of the bookshelf, the properties of a strip-grinding machine constrain each PF coated veneer core plywood segment's geometry. The segments need to be attached to a support at a fixed distance from the strip. Given the minimal length needed to fix the parts to the support and the length from the support to the strip, which depends on the slipping angle, it was decided that the minimal segment length should be 100 mm (lmin) and the smallest angle 33° (αmin).

These requirements are expressed by the following function:

$$\begin{aligned} p_l &= k_l \cdot \min(l) \text{ } if \text{ } \min(l) \leq l_{\min} \\ p_l &= 100 \text{ } otherwise \\ k_l &= 100 / l_{\min} \end{aligned} \quad (2)$$

where min(l) is the shortest cell wall found in an individual and

$$\begin{aligned} p_\alpha &= k_\alpha \cdot \min(\alpha) \text{ } if \text{ } \min(\alpha) \leq \alpha_{\min} \\ p_\alpha &= 100 \text{ } otherwise \\ k_\alpha &= 100 / \alpha_{\min} \end{aligned} \quad (3)$$

where min(α) is the smallest angle measured in an individual.

The chosen plywood thickness is 9 mm, which is overdimensioned, meaning to be stable enough for whatever Voronoi network. Consequently, there was no engineering constraint. As a control, the final bookshelves presented in this paper were subsequently analysed (see the Results section), and showed no excessive deformation. A functional requirement for a bookshelf is how well books stack in the compartments. A critical factor for this is the angle between the lower walls in each compartment. A 90° angle is optimal for books to be stacked in a compartment, an angle β_0 between 80° and 100° is considered acceptable. The individual is therefore scored after its percentage of cells that fulfil this specification.

$$\begin{aligned} p_\beta &= k_\beta \cdot n(\beta) / n(cells) \\ k_\beta &= 100 \end{aligned} \quad (4)$$

The objective function to be maximised is

(5) $$p = p_l + p_\alpha + p_\beta$$

which corresponds to the fitness of each individual. The constants k_l, k_α and k_β have been determined so that the maximum score for fulfilling each of the requirements is the same, in this case 100 points (in other words this is a sum for which all the requirements have the same weight).

Results

Sample bookshelves generated by the designers who tested the interface are presented in Figure 6, showing variations in terms of dimensions and compartments. Fig. 7 shows the fitness curve of the evolution during the optimisation of one of the designer's bookshelves. The population converges towards the feasible solution space; using a population of 50 individuals during 300 generations was satisfactory.

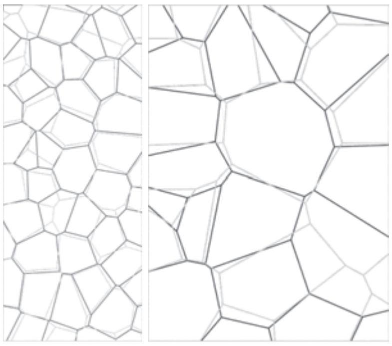


Fig. 6 Examples of bookshelves generated by the designers. The original user defined structure in black; optimised solutions in grey.

Two shelves are presented in more detail. Both (700x2000 mm and 2000x2000 mm, 9 mm plywood) were generated using the application and are shown in **Fig. 4** and **Fig. 8**. Their optimisation took around 1 hour on a dual-core 2.2 GHz processor.

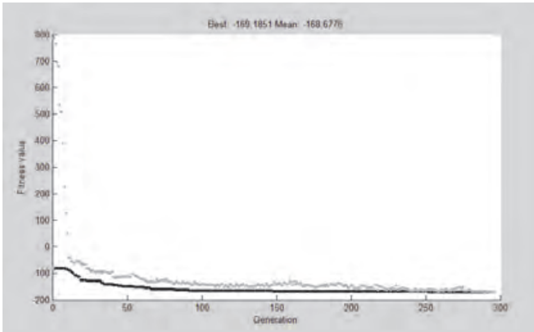


Fig. 7 Typical appearance of the fitness curve during the optimisation. In black: the fitness of the best individual, in grey: the mean fitness of the current population

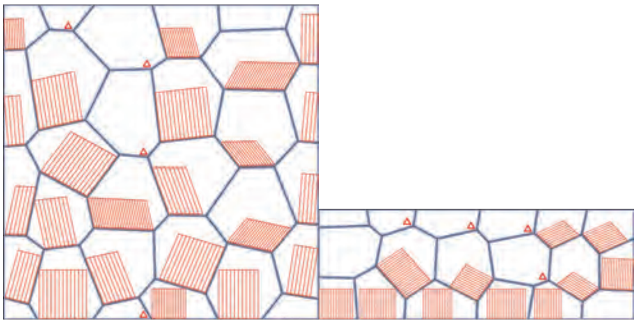


Fig. 8 Illustration of the final bookshelves with stacked books. The triangles indicate where the functional objective (equation 4) has not been met

To verify that these bookshelves were structurally sound and stable, they were tested with FEM-analysis. To simulate the weight of books, the bookshelves were subjected to a load of 10 kg in each compartment as well as the standard gravitational acceleration. The material used was 9 mm Low-density fibreboard (LDF-board, an engineered wood product that has properties similar to PF coated veneer core plywood, but is isotropic, which makes the analysis simpler and more conservative), with the modulus of elasticity being 8 GPa, Poisson's ratio being 0.3 and density being 500 kg/m^3 [29]. The analysis was done in ANSYS Workbench®. The structural analysis indicated that the maximum deformation of either bookshelf never exceeded 0.4 mm. The results are shown in **Fig. 9** and **Table 1**.

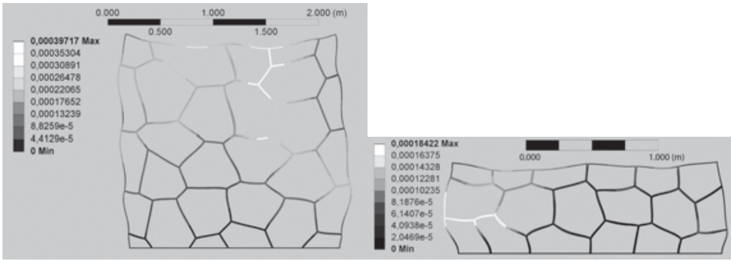


Fig. 9 Structural analysis of the two bookshelves using ANSYS Workbench®

Table 1 Results from the analysis in ANSYS Workbench®

Bookshelf	Volume(m^3)	Weight(kg)	Max. deformation(m)
700x2000	0.0538	26.911	1.84×10^{-4}
2000x2000	0.1225	61.259	3.97×10^{-4}

To verify the aesthetic qualities and their commercial potential, the two bookshelves were subsequently 3D-modelled with 9 mm board thickness; then photo-realistically rendered based on the output geometry of the application and subsequently shown to peer designers. The photo-realistic renderings are shown in **Fig. 10** and were very well received. To confirm the bookshelves' manufacturability, drawings have been produced from the output and two prototypes are being built.

Fig. 10 Photo-realistic renderings of two bookshelves

Conclusion and further research

In this study, a bookshelf generating system using computational geometry (Voronoi structure) – taking into account functional, engineering and production constraints – has been developed. This second application further validates our approach and confirms that using complex forms for designing artefacts has the potential to become a more common practice.

Higher levels of user control, feedback and automation could be implemented in the present and similar applications. In terms of user control, a higher degree of freedom in manipulating the structure would be desirable, for example adding, removing and moving points and defining a custom contour of the bookshelf. Concerning feedback, visual indicators relating to the monitoring of functional and production parameters in the creation and manipulation process could be integrated. Equally important would be to allow users to input individual requirements such as what types of objects are to be stored and in which quantity. Surprisingly, some designers actually desire to relinquish control and let the algorithm determine the design in the full for them. Automated production drawing or data generation would be a useful feature as well as an approximate estimate of cost dependent on material choice, but also weight and sizes of items to be packed and shipped. The interface has been tested by some designers, but the generation of the bookshelf could be equally controlled by consumers to provide them with *bespoke* products.

It is foreseeable that the continuation of this work may result in a generalised tool; that is to say an application that allows designers or consumers to chose from a much wider range of tessellations in order to generate an equally wider range of product types. It is well conceivable that such application will allow for any user-created or user-input tessellations but also integrate entirely new usability, functionality and production constraints for new types of products. This will be the object of future research.

References

1. Nordin A, Hopf A, Motte D, Bjärnemo R, Eckhardt C-C (2009) Using genetic algorithms and Voronoi diagrams in product design. J of Computing and Information Science in Engineering - JCISE, submitted
2. Shea K (2004) Explorations in using an aperiodic spatial tiling as a design generator, 1st Design Computing and Cognition Conference DCC'04, Cambridge, MA, 137-156

3. Friebe H, Ramge T (2008) The DIY Brand - The Rebellion of the Masses against Mass Production (In German. Original title: Marke Eigenbau: Der Aufstand der Massen gegen die Massenproduktion). Frankfurt: Campus Verlag
4. Lakhani KR, Kanji Z (2009) Threadless: The Business of Community, Harvard Business School Multimedia/Video Case 608-707, Harvard Business School, Cambridge, MA
5. Moser K, Müller M, Piller FT (2006) Transforming mass customisation from a marketing instrument to a sustainable business model at Adidas. International J of Mass Customisation 1: 463-479
6. Bouché N (2009) Keynote V: How could we create new emotional experiences with sensorial stimuli?, 4th International Conference on Designing Pleasurable Products and Interfaces - DPPI'09, Compiègne
7. Toffler A (1971) Future Shock. 2nd ed. New York, NY: Bantam Books
8. Hopf A (2009) Renaissance 2.0 - Expanding the morphologic repertoire in design, 23rd Cumulus Conference 23/09, Melbourne
9. McCormack J, Dorin A, Innocent T (2004) Generative design: a paradigm for design research, Futureground, Melbourne
10. Lin B-T, Chang M-R, Huang H-L, Liu C-Y (2009) Computer-aided structural design of drawing dies for stamping processes based on functional features. The International J of Advanced Manufacturing Technology 42: 1140-1152
11. Kraft Foods (2006), Innovate with Kraft, Available online, brands.kraftfoods.com/innovatewithkraft/default.aspx, Last accessed: January 2010
12. Nike Biz Media (2007), New NIKEiD studio opens at NikeTown London giving consumers a key to unlock the world of design, Available online: www.nikebiz.com/media/pr/2007/11/01_nikeidlondon.html#, Last accessed: January 2010
13. NIKEiD (1999), Available online, www.nikeid.com, Last accessed: January 2010.
14. Anderson C (2006) The Long Tail: Why the Future of Business is Selling less of more. New York, NY: Hyperion
15. Knight TW (1980) The generation of Hepplewhite-style chair-back designs. Environment and Planning B 7: 227-238
16. Orsborn S, Cagan J, Pawlicki R, Smith RC (2006) Creating cross-over vehicles: Defining and combining vehicle classes using shape grammars. Artificial Intelligence for Engineering Design, Analysis and Manufacturing - AI EDAM 20: 217-246
17. McCormack JP, Cagan J, Vogel CM (2004) Speaking the Buick language: Capturing, understanding, and exploring brand identity with shape grammars. Design Studies 25: 1-29
18. Pugliese MJ, Cagan J (2002) Capturing a rebel: Modeling the Harley-Davidson brand through a motorcycle shape grammar. Research in Engineering Design 13: 139-156

19. Chau HH, Chen X, McKay A, de Pennington A (2004) Evaluation of a 3D shape grammar implementation, 1st Design Computing and Cognition Conference DCC'04, Cambridge, MA, 357-376
20. Cluzel F, Yannou B (2009) Efficiency assessment of an evolutive design system of car contours, 17th International Conference on Engineering Design - ICED'09, Stanford, CA
21. Orsborn S, Cagan J, Boatwright P (2008) Automating the creation of shape grammar rules, 3rd Design Computing and Cognition Conference DCC'08, Atlanta, 3-22
22. Shea K, Cagan J (1999) Languages and semantics of grammatical discrete structures. Artificial Intelligence for Engineering Design, Analysis and Manufacturing - AI EDAM 13: 241-251
23. Ang MC, Chau HH, McKay A, de Pennington A (2006) Combining evolutionary algorithms and shape grammars to generate branded product design, 2nd Design Computing and Cognition Conference DCC'06, Eindhoven, 521-539
24. Morel P, Hamda H, Schoenauer M (2005) Computational chair design using genetic algorithms. Concept 71: 95-99
25. Wenli Z (2008) Adaptive interactive evolutionary computation for active intent-oriented design, 9th International Conference on Computer-Aided Industrial Design and Conceptual Design - CAIDCD'08, Kumming
26. de Berg M, Cheong O, van Kreveld M, Overmars M (2008) Computational Geometry - Algorithms and Applications. Berlin: Springer-Verlag Berlin Heidelberg
27. Franke N, Piller F (2004) Value creation by toolkits for user innovation and design: The case of the watch market. The J of Product Innovation Management 21: 401-415
28. Ferrum Lasercut GmbH, personal communication
29. Wood Handbook - Wood as an Engineering Material (1999) Madison, WI: Forest Products Laboratory

Paper IV

Strategies for consumer control of complex product forms in generative design systems

IV

Nordin, A., Motte, D., & Bjärnemo, R.

39th Design Automation Conference - DETC/DAC'13,
Portland, OR, August 4-7, 2013

Originally published by ASME

Proceedings of the ASME 2013 International Design Engineering Technical Conferences &
Computers and Information in Engineering Conference
IDETC/CIE 2013
August 4-7, 2013, Portland, Oregon, USA

DETC2013-12886

STRATEGIES FOR CONSUMER CONTROL OF COMPLEX PRODUCT FORMS IN GENERATIVE DESIGN SYSTEMS

Axel Nordin*, Damien Motte, Robert Bjärnemo
Division of Machine Design
Department of Design Sciences LTH
Lund University
Box 118, 221 00 Lund
Sweden
Email: axel.nordin@mkon.lth.se,
damien.motte@mkon.lth.se,
robert.bjarnemo@mkon.lth.se

ABSTRACT

In recent years, the number of products that can be tailored to consumers' needs and desires has increased dramatically; there are many opportunities to individualize the colors, materials or options of products. However, current trends indicate that the future consumer will not be satisfied with mere material and color choices, but will desire control over form as well. While it is technically feasible to allow consumers to partially mass-customize the form of products subject to functional and production constraints through the use of a generative design system, the question of how the control of form should be presented to the user arises. The issue becomes especially important when the product form is based on complex morphologies, which require in-depth knowledge of their parameters to be able to control them fully. In this paper, we discuss this issue and present and test two strategies for controlling complex forms in consumer-oriented generative design systems, one offering the user full control over the design ("total control" strategy), while the other automatically generates designs for the user ("no control" strategy). The implementation of those two control strategies in a generative design system for two categories of products (bookshelf and table) and five types of morphologies are described and tested with a number of design interested participants to estimate their level of satisfaction with the two control strategies. The empirical study shows that the participants enjoyed both the total control and no control strategies. The development of the full control modes for the five morphologies was on the other hand not straightforward, and in general, making the controls meaningful to the consumer can be difficult with complex morphologies. It seems that a consumer-oriented generative design system with two different control strategies, as the ones presented in this article, would offer the most satisfaction.

INTRODUCTION

In recent years, the number of products that can be tailored to consumers' needs and desires (known as mass-customization) has increased dramatically; there are many opportunities to individualize the colors, materials or options of products. Sooner than later, the future 'prosumer' [1] will not be satisfied with mere material and color choices, but will desire control over form as well. This is illustrated by the interest showed for companies such as Shapeways (http://www.shapeways.com/) and Materialise (http://www.materialise.com/) that offers their customers products such as customizable decorative items or user-designed add-ons for existing products manufactured with rapid prototyping techniques which enable one-off production and very intricate forms. Such techniques, however, can currently only be used for certain types of consumer products as it has high costs in comparison with traditional production methods and is limited regarding materials, and product size.

* Address all correspondence to this author.

Traditional production methods, on the other hand, require the product form to comply with technical constraints specific to the production equipment and thus limit the possibilities for form customization.

In the case where the technical and functional constraints are well-defined, it is however possible to partially mass-customize the shape of products that are produced with traditional production systems, through the use of so-called "generative design systems". A generative design system is basically structured around a graphical user interface with which the user can modify and evaluate product forms, and an interactive optimization system or an interactive constraint satisfaction system that handles user and technical preferences and constraints. Most generative design systems have been intended for professional designers, for example to help the designer preserve the "form" identity of a brand [2-5]. Only a few have been developed for consumers, e.g. Kram/Weisshaar developed the Breeding Tables program, which generates variations of a table design using a genetic algorithm that modifies a set of parameters ruling the support structure [6].

For this category of generative design systems intended to be used by consumers comes the question of the control of form, especially when complex morphologies such as natural-mathematical forms are used. These morphologies are for example the minimal surfaces or cellular automata (Figure 1). Many morphologies require advanced knowledge of their parameters to be able to control them fully, which is out of range of virtually all consumers. How can such morphologies then be used when the user has limited or no knowledge in programming or mathematics? The problem becomes even more complex when he or she has to take into account technical and functional constraints.

In this paper, we discuss this issue and present and test two control strategies for controlling complex forms in consumer-oriented generative design systems. The issue of control is first discussed and the two control strategies are presented. The implementation of those control strategies in a generative design system is then described for two categories of products (bookshelf and table) with different manufacturing techniques and five types of morphologies. The empirical study of those control strategies is then presented, and consequences in relation to mass customization are discussed.

CONTROLLING COMPLEX FORMS

There are several possible ways to control complex forms, some requiring a lot of manipulations from the users, some requiring virtually none. The work from Piasecki and Hanna [7] is useful here to provide some guidance. Piasecki and Hanna [7] address the issue of control in relation to complexity and consumer satisfaction. They build on the research done by Schwartz [8] who defined the so-called "paradox of choice": a large amount of choice among product alternatives is positively correlated with consumer satisfaction, but too much choice can lead to dissatisfaction and confusion. They showed that there exists an optimal amount of choice, but that it is not only the amount of choice but also the amount of meaningful choice that influences consumer satisfaction. Expressed in another way, the consumer may want to have a lot of control of the morphologies but only if the way the morphologies are controlled helps him or her achieving his or her goals (such as sufficient spaces for books in the case of a bookshelf, furniture volume that fits the room it is intended for, etc.).

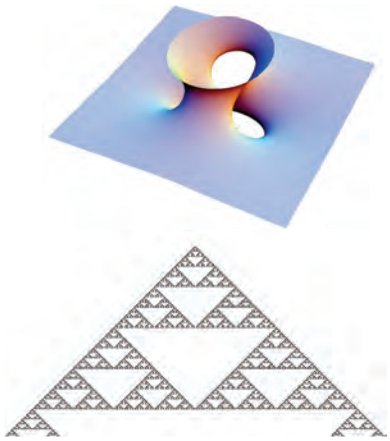


Figure 1. Examples of mathematical morphologies. Top: minimal surface [9], bottom: cellular automata [10]

Therefore, having complete freedom of creation, such as in traditional parametric CAD or surface modeling tools such as Autodesk Alias, is not particularly adapted to consumer-oriented generative design systems. Such software have a steep learning curve and can imply a large workload for relatively simple tasks.

Another possibility is to have a partial control strategy. The user is allowed to design a rough form for the product with a simplified interface. When the user is satisfied with his or her preliminary design, the system tries to satisfy the technical and functional constraints and objectives while remaining as close as possible to the user's original design. A study of this strategy was performed with professional designers [11]. The results showed that this control strategy was not particularly appreciated for several reasons. Some pointed out that it was frustrating that the generative design system altered, even a little bit, their design. For complex forms (probably even simple ones) the least changes can give a completely different visual expression to the product. This study indicated also that one part of the respondents would prefer maximum control of the form creation process (which would include dealing manually with the constraints), the other part — rather surprisingly for professional designers — wanted a generative design system that would generate the product completely automatically before they could choose their preferred design. Although these two different potential control strategies come from professional designers, they should still be relevant as a starting point for devising the morphology control strategies for consumers. Moreover, these two control strategies present strengths and weaknesses that could be beneficial for further investigation.

The aim of this paper is consequently to investigate these two alternative strategies, hereafter called the "total control" and "no control" strategies. These control strategies are discussed below.

The total control strategy

The total control strategy is based on leaving as much of the control of the morphology to the user as possible, while the generative design system can assist him or her with information about the evaluation of the form in terms of manufacturability and functionality. It is therefore not full control in the sense of traditional CAD software, but rather that the user is free to manipulate the form within the limits of the morphology. This presents several benefits. This strategy removes the need for implementing an optimization system as it is done manually by the consumer. Moreover, it has been shown that allowing consumers to personalize their products in some manner increases their attachment to the product, and increases the perceived value of the product [12].

The concept of total control raises also several issues. First, as mentioned earlier, most users do not possess the skill to manipulate the morphologies directly through their mathematical definition, which prevents *de facto* complete control. It is necessary to have a specific interface for each type of morphology. It also presents the challenge to implement an interface for the consumer that has enough freedom of creation to satisfy the consumer, while hiding irrelevant aspects, and giving instant feedback of important properties such as producibility, stability and weight It is difficult to assess to what extent the consumer wants or needs to be active (cf. the paradox of choice above). Moreover, users may want to use the morphology as a starting point and then alter it by defying the original definition of the morphology to suit their own preferences, for example by adding and removing curves, lines and vertices in the structure. This presents complex computational problems: if the elements of the morphologies are linked by specific equations, how then should changes introduced by the user be handled? We have consequently interpreted the total control requirement as the possibility to manipulate in a detailed way the elements of a morphology while remaining true to its underlying natural-mathematical structure.

The no control strategy

In the no control strategy, the user lets the design generation system generate solutions that fulfill engineering and production constraints without intervening in the process. The user is still able to define certain design parameters such as the outline of the product (shelf or table), or height or depth, but the detailed control over the morphology is left entirely to the design generation system. This strategy hides the complexity of handling the morphology from the user, and does not require an interface for handling the morphologies to be implemented, but the automated handling of constraints and optimization must be implemented. It may also be necessary to generate several candidate solutions so that the user is able to make a choice. The constraint satisfaction process might require a noticeable amount of time for the system to give valid solution proposals, and thus might be a cause of frustration for the user. In works such as those of Sims [13] and Secretan [14] the user is active in the selection and evaluation of the solution candidates while the optimization is running in order to guide the optimization in terms of hard-to-quantify criteria such as aesthetics. In order to reduce frustration and user fatigue, techniques such as those developed by Ren [15] implement an active learning algorithm to propose only promising designs to the user. These techniques however require the user to interact several times with the generative design system, which is problematic for design problems that require many iterations or have long evaluation times. Such techniques may be appropriate for professional designers, but is felt too demanding for average consumers. Because of this, the technique used in this paper is based on generating the solutions without user input, and instead lets the user make a design selection only among the final feasible solutions.

Nevertheless, if the no control strategy experience is as much appreciated as the total control strategy, this would be an attractive control alternative in the development of consumer-oriented generative design systems as it removes the need for the development of potentially complex interfaces for the manipulation of different morphologies.

IMPLEMENTATION OF THE TWO CONTROL STRATEGIES IN A GENERATIVE DESIGN SYSTEM

To be able to test the two control strategies with consumers, they need to be implemented into a generative design system. This section describes the system and its implementation.

The design problem

The generative design system allowed the user to develop either a bookshelf or a support structure of a table in sheet steel or plywood. He or she could also define the contour of the product and the depth of the bookshelf or height of the table. The design problem is such that any design-interested consumer can relate to it, while it is still tied to a set of constraints and objectives that are common to most consumer products. The possible morphologies that the user could choose from (Voronoi diagram, Chinese lattice and isohedral tessellations) are described in depth in the next section. Images of the products based on Voronoi diagrams are represented Figure 2. The design of the support structure of the table and the form of the bookshelf had to comply with different technical constraints.

The table needed to be able to support a weight of at least 50 kg and no part of the table can have a vertical displacement larger than 2.5 mm. The shelves had to be able to support a weight of 10 kg in each compartment. For the shelves there was also a functional objective consisting in the lowest angle of each compartment being in the range of 80° to 90° to aid better stacking of books. The production methods available for sheet metal were bending and laser cutting. The limitations of the bending machine introduced two production constraints. The geometry of the machine limits the flange lengths of the cells of the support structure never to be shorter than 30 mm, and the bending angles never to be less than 35°. The production constraints for plywood were based on the limitations of the saw, which required the pieces to be equal to or longer than 100 mm, and have no cuts less than 35°, as seen in Figure 3.

Note that prototypes could successfully be built according to the production instructions output by the generative design system, verifying that the system was properly functioning [11;16].

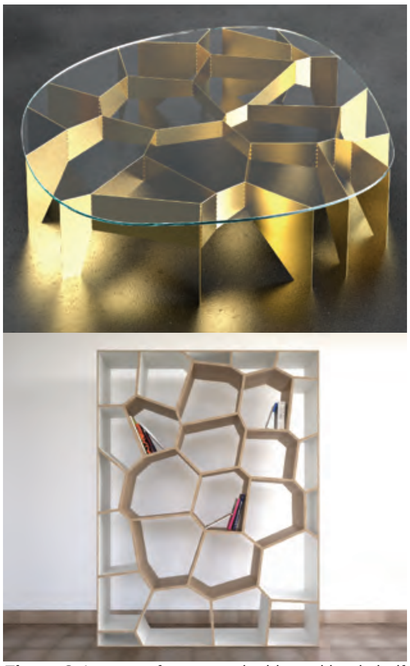


Figure 2. Images of generated table and bookshelf proposals

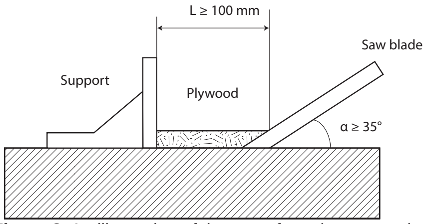


Figure 3. An illustration of the manufacturing constraints

Implementation of the whole generative design system

The generative design system was devised as follows. The interface was divided into five separate steps between which the user could navigate. The first step consisted in choosing which product category to design. This choice affects the later selection steps and evaluation functions. The second step consisted in defining a number of user requirements, such as the height of the table, or depth of the shelf, as well as being able to define the contour of the shelf or table. The contour was defined by drawing a rectangular or polygonal two-dimensional outline (see Figure 4). The third step allowed the user to select what material the product should be made from. The material chosen, and production method linked to it, also affects the evaluation functions, especially the producibility evaluation. The fourth step allowed the user to select a morphology he or she was interested in, and the last step allowed the user to manually manipulate the structure (total control strategy), or launch an optimization (no control strategy).

The morphologies and the manners in which they can be controlled (hereafter, manipulation modes) are presented in the next section.

The whole generative design system and the control strategies have been developed in Matlab and the structural evaluation was performed with the finite element toolbox CALFEM [17].

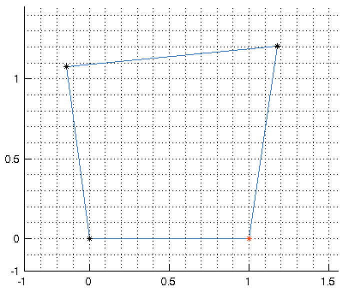


Figure 4. The contour definition interface

Implementation of the total control strategy

The total control strategy allowed the user to move the control points of the morphology while viewing the resulting structure in two or three dimensions. It also allowed for visual feedback of the level of constraint satisfaction in terms of production issues, such as too sharp angles or too short lengths, shown in red, or the amount of deformation of the structure when subject to the loads described earlier. The user could also view detailed information about the structure, such as amount of material used, weight, and estimated cost. Figure 5 presents the visual feedback of production and structural evaluation for a bookshelf with a Voronoi diagram. As can be seen, the pieces of plywood that are too small (less than 100 mm) are represented with a red line, and the angles that are too small (less than 35°) with a red dot. The user can re-arrange the shelf's configuration as much as he or she wants, but is required to ensure that all constraints are fulfilled.

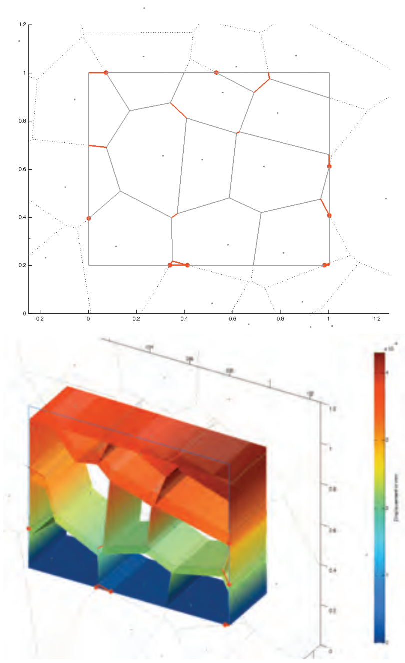


Figure 5. The visual feedback of production and structural evaluation for a bookshelf with a Voronoi diagram

Implementation of the no control strategy

For the no control strategy, a constrained optimization system was utilized which took the manufacturing and functional constraints and objectives into account, and offered the user a design suggestion after it had finished. The generation of design concepts took the generative design system around 15 minutes.

For the automated generation of the valid design solutions, the constraints were converted to a single "objective" through the use of a weighted sum of the constraint violations by the genetic algorithm (GA) implementation provided by Matlab using the coordinates of the form control points (see the next section) as the genome, a population of 50 individuals – and the stop criterion being that all constraints should be satisfied (i.e. the weighted sum of the constraint violations should be 0). The full algorithm is described in length in [16].

DESCRIPTION OF THE MORPHOLOGIES

In this section the morphologies used in the generative design system and their manipulation modes (how the morphology can be controlled by the user or constrained optimization system) are described. The morphologies have been chosen to represent both unconstrained and complex morphologies (the Voronoi diagram and the Chinese lattice), and more straightforward morphologies with few possibilities of manipulation (three isohedral tessellations).

Voronoi diagrams

A Voronoi diagram (Figure 6) can be described as follows: Let p_1,.., p_n be a set of n distinct points in the plane; these points are called the Voronoi sites. For each site, the set of points that are closer to it than to any other site form a Voronoi cell. A Voronoi diagram consists of all such cells. An overview of the properties of a Voronoi diagram can be found in [18, chapter 7]. The Voronoi diagram itself is manipulated by moving, adding, or removing the points p_1,.., p_n.

Even with relatively few Voronoi points the diagram becomes complex to handle. A small change in the position of one point can greatly affect the surrounding cells. This makes it a good morphology for testing how users respond to complex morphologies, which they have not previously come in contact with.

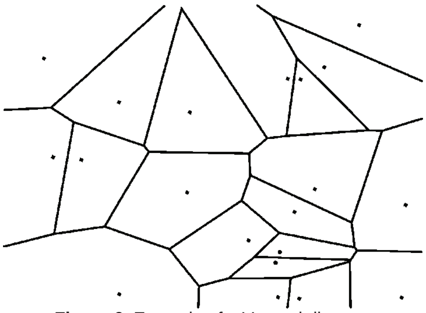


Figure 6. Example of a Voronoi diagram

Chinese lattice

The Chinese lattice structure is basically generated by dividing any polygon into two new polygons (see Figure 7). A thorough description of the generation of Chinese lattices can be found in [19]. The structure is manipulated by adding, removing or moving two points for each bisecting line to define its position and direction.

Similarly to the Voronoi diagram, this morphology can become quite complex to handle if many points are added. The logic of how the morphology is constructed might however be more intuitive than the Voronoi diagram, which makes it suitable for testing how complex, yet intuitive morphologies are perceived by a user.

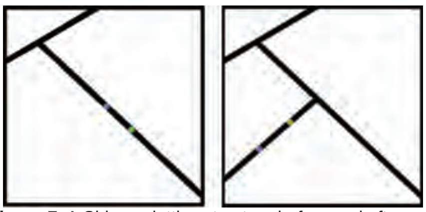


Figure 7. A Chinese lattice structure before and after one bisecting operation

Isohedral tessellations

The isohedral tessellations used in the application are two-dimensional and tile the Euclidian plane. An isohedral tiling consists of polygons surrounded by copies of themselves. There exist 42 unique isohedral tessellations consisting of symmetric polygons. An in-depth description of isohedral tessellations can be found in [20].

Three of these have been implemented in the application, the pentagonal D1, the hexagonal D1, and the kite tessellation, see Figure 8. The isohedral tessellations used in the application are manipulated by moving predefined vertices in the original polygon. It is not possible to remove or add vertices to the polygon.

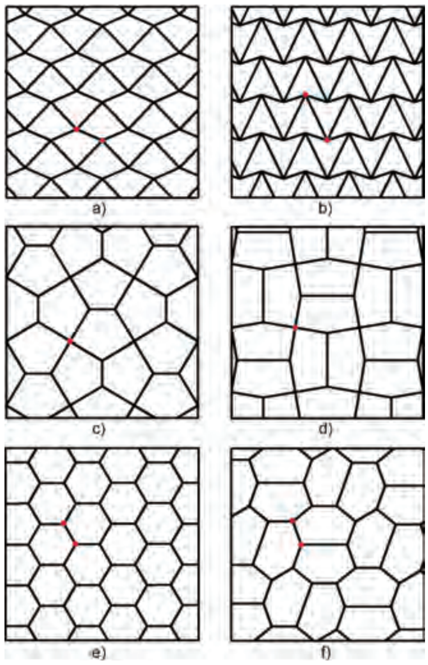


Figure 8. Two instances per morphology resulting from different locations of the control points (shown in red) are represented. a) and b) show the Kite morphology, c) and d) show the Pentagonal D1 morphology, and e) and f) show the Hexagonal D1 morphology

The mathematical formulations of the chosen isohedral tessellations allow for either one control point (as for the pentagonal D1 tessellation) or two control points (as for the Kite and Hexagonal D1 tessellations). The low number of control points makes the manual handling of the morphologies quite straightforward for the user in comparison to the Voronoi diagram and Chinese lattice.

THE STUDY

Aims

The study of the full control and no control strategy was decomposed into four aims:

1. Determine the satisfaction level for each control strategy

2. Determine whether one strategy is significantly more appreciated than the other
3. For the full control strategy, determine the satisfaction level of the defined morphology manipulation modes
4. For the no control strategy, estimate an acceptable waiting time for design generation

The test set-up and procedures

Basic set-up and procedure

To evaluate the two control strategies a test station was set up at a design exhibition center (Form/Design Centrum in Malmö, Sweden) attracting visitors with a strong interest in design and furniture. The set-up consisted in the display of one shelf and one table generated with the program, as well as general information about the application (see Figure 9). Individual visitors of the center were asked to participate in designing their own shelf or table using the application. The participants were diverse in terms of computer experience and aesthetic training, as well as age and gender. We chose the location and participants to ensure that the persons using the application would have an interest in buying design-oriented furniture. After a quick demonstration they were asked if they would like to use the total or no control mode of the application. They were guided through the settings and usage of the application, and could then individually spend any amount of time using the program. The volunteers desiring to test the system could either use the total control or no control strategy. It was chosen not to let the participants evaluate both systems in order to avoid frustration, fatigue or boredom which would affect their experience with the system. The participants were asked to fill out a questionnaire regarding their satisfaction with the design system immediately after they had finished their design task. In the case of the no control strategy the participant was asked to come back after 15 minutes to review the design proposed by the system, and then fill out the questionnaire.

Figure 9. Experimental set up at Form/Design Center

Design of the questionnaire

A questionnaire was selected as the method of evaluating the user's satisfaction with the generative design system. Other alternatives such as ”thinking aloud” [21], co-discovery [22, p. 198] or interviews, were also considered but to get quantifiable results from the participants a questionnaire based on the visual analogue scale was selected. The visual analogue scale is relatively straightforward and quick for the participant to fill in and gives results which are easy to handle statistically.

Test procedure for the comparison of the total control and no control strategies (aims 1 and 2)

For determining the overall satisfaction level for each control strategy (aim 1), each participant was asked to evaluate his or her level of satisfaction with the system on a visual analogue scale, from 0 (worst possible experience) to 1 (best possible experience), as well as the satisfaction with the obtained design.

For determining whether one control strategy was preferred (aim 2), two tests were realized. First the users could spontaneously choose the type of control strategy they wanted to use. This would indicate that one control strategy is *a priori* more attractive that the other. The null hypothesis was that the number of users that would choose one control strategy would not be significantly larger that the number of users that would choose the other control strategy. Second, the satisfaction level for each control strategy could give an indication of whether the experience of one control strategy was better than the other. The second null hypothesis was that the satisfaction levels of both control strategies were not significantly different.

The difficulty for the second strategy was to establish a relevant “significant difference between satisfaction levels.” It was estimated that a difference of less than 15% between the two control strategies would give no ground for deciding in favor of one over the other. Moreover, the standard deviation was supposed to be at most 15% (if one supposes a normal distribution around a mean of .50, a standard deviation (SD) of .20 means that 95% of the participants' evaluations are predicted to be between .20 and .80). This would give an estimated size effect of .15 / .15 = 1.00. With this estimation, it was also possible to determine the necessary number of participants. A power estimate of .80 was decided to be satisfactory, and to reach that level of power with the estimated effect size, 16 participants in each of the two groups were needed.

Test procedure for the total control strategy (aim 3)

The visitors who chose to use the full control strategy were asked to estimate his or her level of satisfaction with the manipulation mode of each tested morphology on a visual analogue scale with scores ranging from 0 (not satisfied at all) to 1 (extremely satisfied).

Test procedure for the no control strategy (aim 4)

After the participants obtained the generated results, they were asked to estimate how much time they were willing to wait for the design to be generated (from 0 to 24 hours) on a visual analog scale.

Results

Comparison of the total control and no control strategies (aims 1 and 2)

Seventeen participants spontaneously chose the total control strategy and 9 participants chose the no control strategy. A double-sided binomial test showed that there was no significant preference for one strategy over the other ($p = .08$). In order to complete the test, 7 more participants used only the no control strategy, these participants did not belong to exactly the same group as the 9 participants that freely chose the no control group and this might have affected the test result. However, they were not made aware of the total control strategy and the mean of these 7 participants' level of satisfaction was actually higher than that of the self-selected participants. In total, 17 participants tested the total control strategy and 16 the no control strategy. In the first control strategy, the mean (M) satisfaction score was M = .77 ($SD = .15$), in the second control strategy M = .84 ($SD = .13$). The t test of the difference between means did not produce a statistically significant result ($p = .13$).

Each participant was also asked whether he or she was satisfied of his or her final design. For the total control strategy, the mean score was M = .70 ($SD = .27$), and for the no control strategy M = .82 ($SD = .11$). The t test of the difference between means did not produce a statistically significant result either ($p = .11$).

Total control strategy: Evaluation of the manual handling of the morphologies (aim 3)

Seventeen visitors chose to use the full control strategy. The Voronoi diagram got a mean score $M = .83$ ($SD = .21$) for a number (N) of $N = 10$ participants, the Chinese lattice a score of $M = .83$ ($SD = .20$, $N = 10$), the D1 pentagon tessellation $M = .79$ ($SD = .28$, $N = 10$), the D1 hexagon tessellation $M = .83$ ($SD = .18$, $N = 10$), and the kite morphology $M = .83$ ($SD = .16$, $N = 12$). The way the morphologies could be manipulated was by and large appreciated by all but a few participants (which explains the large SD for the D1 pentagon tessellation). There were no signs of frustration regarding the expressed limitations of the morphologies manipulated, unlike in the case of the partial control strategy [11].

No control strategy: Estimation of acceptable waiting time for design generation (aim 4)

Fifteen of the 16 participants answered this question. The average response was a waiting time of 20 h and 19 min ($SD = 7{:}19$, $N = 15$). One participant did not want to wait much more than half an hour (36 min) and another 4 h 36 min; all the remaining participants were willing to wait more than 20 h, some more than one day, with an average of 23 h 3 min ($SD = 1{:}06$, $N = 13$).

DISCUSSION

The total control and no control strategies, with satisfaction scores of .77 and .84 respectively, were equally appreciated by the participants, unlike the previously tested partial control strategy [11].

For the total control strategy, the different morphologies, with their different levels of complexity, were equally appreciated. The users of the total control strategy were also able to cope with the complex nature of some of the morphologies, and manipulate the tessellations to satisfy the constraints, based on the visual feedback, even though there were many parameters that could be adjusted. This seems to indicate, based on the research of Piasecki and Hanna [7], that the manipulation modes were both meaningful and rather intuitive to the user. The fact that morphologies of different natures were equally appreciated is a first step towards a generalization of the results although there is large leap from 2D morphologies to 3D or even 4D (dynamic) morphologies. Finding relevant and meaningful morphologies manipulation modes is however not straightforward (several alternative modes were originally devised for the Voronoi diagram) and rather time-consuming. This is an important factor to take into account in the development of such consumer-oriented generative design systems.

In the no control strategy, although the user could not influence the final result much, the perception was still that the product is tailored to individual needs and expectations. None of the participants expressed the desire to have a choice among several design proposals which is important given the time an automated design generation might require. The automatic design generation took around 15 minutes to complete. Although there might exist more efficient ways of implementing the optimization system described in this paper, the optimization of other, more complex products might still require a noticeable amount of time to finish. In this respect, a generation time of 15 minutes was deemed adequate as it did not give instantaneous results, but was fast enough so that the participants could review the results during their visit. Additionally, according to the questionnaire-answers, the participants were willing to wait almost a day to get results. This is a surprisingly high value, a finding that is counterintuitive within human-computer interaction research. This would allow much more freedom for the elaboration of any generative design system. This is to be compared to the long waiting times usually encountered by consumers when dealing with craftspeople and companies offering bespoke products (meaning: custom-made and built-to-order). If the consumer is certain of receiving a satisfying result where no further manipulation is needed, a long waiting time is not negatively perceived. However, two participants using the total control complained that the time needed to compute and display the feedback of manufacturing and stability issues (only 1-2 seconds) was too long in the full control strategy, which tends to show that the acceptable waiting time is strongly dependent on the number of iterations between user and software that are needed to achieve a satisfying result.

The participants were also satisfied by their final designs. Such results must be handled with caution as the participants were by no means in a real design situation — where the finalized design would be put into production. Moreover, the different dimensions constituting the overall satisfaction (e.g. novelty) have not been investigated. But once again, the respondents did not present signs of dissatisfaction regarding their tessellations-based designs, which is a first positive result towards the use of complex morphologies in design.

Although the questionnaire was anonymous, different biases may have occurred. For example, the participants may have overestimated their satisfaction level as “a sign of encouragement” towards the developers of the system. Nevertheless, it is likely that the participants were not unsatisfied, or would have expressed their dissatisfaction, as the designers did regarding the partial control strategy. An aspect that has not been accounted for is the effect of training on the satisfaction level. A trained user would perhaps have a lesser level of satisfaction than an occasional or single-time user.

In this paper, we have investigated how 2D morphologies can be applied to furniture as it provides a suitable test bed because any person understands what furniture represents, and it is still constrained in terms of weight, stiffness and visual appeal. However, the approach should be possible to apply to other, more complex, product typologies and morphologies as well, and current studies are directed at exploring its domain of applicability.

CONCLUSION

The handling of complex morphologies is not straightforward, but the users seem to enjoy both total control and no control strategies. It seems that a solution with two different modes, as the one presented in this article, offers the user the most satisfaction, as opposed to a solution with only one in-between mode, as presented in [11]. At the outset of this article, the trend towards consumer participation in product design and realization was presented. The fact that there was no sign of dissatisfaction for the no control strategy implies that Toffler’s ‘prosumer’ [1] not necessarily desires to be deeply involved in the intricacies of algorithmic design to experience new heights of empowerment. Moreover, the fact the users were ready to wait more than 20 h in average implies a large flexibility for online solutions (possibility to defer the calculation part to a remote server, possibility to queue and handle the requests in different ways, etc.).

REFERENCES

[1] Toffler, A., 1971, *Future Shock*, 2nd Edition, Bantam Books, New York, NY.

[2] Chau, H. H., Chen, X., McKay, A. and de Pennington, A., 2004, "Evaluation of a 3D shape grammar implementation", *1st Design Computing and Cognition Conference - DCC'04*, Cambridge, MA, pp. 357-376.

[3] EZCT Architecture & Design Research, Hamda, H. and Schoenauer, M., 2004, *Studies on Optimization: Computational Chair Design using Genetic Algorithms*, EZCT Architecture & Design Research, Paris.

[4] McCormack, J. P., Cagan, J. and Vogel, C. M., 2004, "Speaking the Buick language: Capturing, understanding, and exploring brand identity with shape grammars", *Design Studies*, **25**(1), pp. 1-29.

[5] Pugliese, M. J. and Cagan, J., 2002, "Capturing a rebel: Modeling the Harley-Davidson brand through a motorcycle shape grammar", *Research in Engineering Design*, **13**(3), pp. 139-156.

[6] Kram, R. and Weisshaar, C., 2003, *Breeding Tables*, <http://www.kramweisshaar.com/projects/breeding-tables.html>, last accessed: 22 February 2012

[7] Piasecki, M. and Hanna, S., 2010, "A redefinition of the paradox of choice", *4th Design Computing and Cognition Conference - DCC'10*, Stuttgart, pp. 347-366.

[8] Schwartz, B., 2004, *The Paradox of Choice: Why More is Less*, HarperCollins Publishers, New York, NY.

[9] Metlin, 2008, *Costa Minimal Surface [image]*, <http://commons.wikimedia.org/wiki/File:Costa_minimal_surface.jpg>, last accessed: 20 January 2013. CC BY-SA 3.0.

[10] Landis, N., 2008, *Cellular Automata [image]*, <http://commons.wikimedia.org/wiki/File:R090_pulse_wide.png?uselang=sv>, last accessed: 20 January 2013. CC BY-SA 3.0.

[11] Nordin, A., Motte, D., Hopf, A., Bjärnemo, R. and Eckhardt, C.-C., 2010, "Complex product form generation in industrial design: A bookshelf based on Voronoi diagrams", *4th Design Computing and Cognition Conference - DCC'10*, Stuttgart, pp. 701-720.

[12] Mugge, R., 2007, *Product Attachment*, PhD thesis, Dept of Product Innovation and Management, Faculty of Industrial Design Engineering, Delft University of Technology, Delft.

[13] Sims, K., 1992, "Interactive evolution of dynamical systems", *First European Conference on Artificial Life*, Paris, Toward A Practice Of Autonomous Systems, pp. 171-178.

[14] Secretan, J., Beato, N., D'Ambrosio, D. B., Rodriguez, A., Campbell, A., Folsom-Kovarik, J. T. and Stanley, K., 2011, "Picbreeder: A case study in collaborative evolutionary exploration of design space", *Evolutionary Computation*, **19**(3), pp. 373-403.

[15] Ren, Y. and Papalambros, P. Y., 2011, "A design preference elicitation query as an optimization process", *Journal of Mechanical Design*, **133**(11), pp. 111004-1-111004-9.

[16] Nordin, A., Hopf, A., Motte, D., Bjärnemo, R. and Eckhardt, C.-C., 2011, "Using genetic algorithms and Voronoi diagrams in product design", *Journal of Computing and Information Science in Engineering*, **11**(011006).

[17] Austrell, P. E., Dahlblom, O., Lindemann, J., Olsson, A., Olsson, K.-G., Persson, K., Petersson, H., Ristinmaa, M., Sandberg, G. and Wernberg, P.-A., 2004, *CALFEM : A Finite Element Toolbox - Version 3.4*, Structural Mechanics LTH, Lund University, Lund.

[18] de Berg, M., Cheong, O., van Kreveld, M. and Overmars, M., 2008, *Computational Geometry - Algorithms and Applications*, 3rd Edition, Springer-Verlag, Berlin.

[19] Stiny, G., 1977, "Ice-ray: A note on the generation of Chinese lattice designs", *Environment and Planning B*, **4**(1), pp. 89-98.

[20] Schattschneider, D. and Dolbilin, N., 1998, "One corona is enough for the Euclidean plane", in Patela J. (Ed.), *Quasicrystals and Discrete Geometry*, 1998, American Mathematical Society, Providence, RI, pp. 207-246.

[21] Ericsson, K. A. and Simon, H. A., 1993, *Protocol Analysis - Verbal Reports as Data*, Rev. Edition, MIT Press, Cambridge, MA.

[22] Nielsen, J., 1993, *Usability engineering*, New ed. Edition, AP Professional, Boston.

Paper V

Generative design systems for the industrial design of functional mass producible natural-mathematical forms

V

Nordin, A., Hopf, A., & Motte, D.

5th International Congress of International Association of Societies of Design Research - IASDR'13, Tokyo, August 26-30, 2013, pp. 2931-2941.

Originally published by IASDR

Generative Design Systems for the Industrial Design of Functional Mass Producible Natural-Mathematical Forms

Axel Nordin*, Andreas Hopf**, Damien Motte***

** Division of Machine Design, Department of Design Sciences, Lund University, Sweden, axel.nordin@mkon.lth.se*

*** Division of Industrial Design, Department of Design Sciences, Lund University, Sweden, andreas.hopf@design.lth.se*

**** Division of Machine Design, Department of Design Sciences, Lund University, Sweden, damien.motte@mkon.lth.se*

Abstract: Nature provides us with a vast source of inspiration. However, given industrial designers' open-mindedness and inquisitiveness, a surprisingly limited set of nature-derived symbols continues to be popular in this creative discipline. Rather than designing products mimicking nature, it is probably more rewarding designing them based on the natural principles leading to its growth and form. However, the constraints related to mass produced products make designing with the often complex forms found in nature a daunting task for a human designer. In this paper, we demonstrate, through the implementation of two generative design systems, how fairly complex everyday objects based on three-dimensional natural-mathematical morphologies can be designed, evaluated and produced using mass production techniques; that digital and analogue methods can be linked to create an aesthetic and functional whole beyond purely decorative mimicry. The output from the generative design system made it possible to produce a fully developed, "ready-for-sale" product, with potential for large-scale production. This is a step towards enabling industrial designers the same level of form articulation as has been available to artists and architects, even though the constraints on the design activity are much different.

Key words: Generative design system, L-system, minimal surface, genetic algorithm

1. Introduction

Nature is, and always has been a source of inspiration for human artistic endeavors. This is not without its reasons as all organisms are both image and result of evolution, their forms being the diagrams of invisible natural forces, as Thompson [28] concluded. However, given industrial designers' open-mindedness and inquisitiveness, a surprisingly limited set of deficitary symbols (leaf, tree, hexagon, the colors green and blue, etc.) continues to be popular in this creative discipline. In many cases, unfortunately, nature-inspired design serves as specious greenwashing ingredient in marketing strategies, bestowing a sustainable aura on sometimes very unsustainable products and services. Rather than designing a roof mimicking a snowflake, it is probably more rewarding designing it based on the natural principles leading to its growth and form.

Within design research, the use of morphologies (i.e. forms, shapes or structures) derived from nature has been the object of study for quite some time (see e.g. [22]), however, it has not received the same amount of attention as in for instance art or architecture. The reasons might partly be found in differences in attitudes between the fields, but more importantly in the technical constraints unique to product design. Whereas architectural or artistic projects implementing complex morphologies are most often built in one exemplar with already defined customer and financial resources, the reality for a product based on similar principles is vastly different. Unless the objective of the industrial designer is to create a one-off art-design to be sold to one or very few customers, the design needs to conform to mass production systems, and reliably perform its function. These constraints make the use of complex morphologies, such as the Voronoi-diagram, minimal surface or Lindenmayer systems in design a daunting task. Already time-consuming manual considerations of for instance producibility and structural stability of the product become insurmountable if the form is complex. Although computational methods exist for analyzing arbitrary geometry, they require a highly specialized set of skills which the industrial designer rarely possesses. Instead, industrial design teams in larger companies rely on the expertise of design engineers to evaluate their designs. While this specialization is most often non-problematic, a design project involving complex forms might require many iterations between the departments, leading to increased development time and cost. Having many iterations might also hinder creativity, as trying radically new ideas is associated with high costs.

[19] and [20] suggest using an *interactive generative design system* to aid the designer in the use of complex morphologies. The handling of the form is integrated with computational evaluation systems and constraint handling systems which automatically can steer the form creation towards a solution which is functional, structurally stable, and producible. This approach has been applied to 2.5D objects such as bookshelves and tables using 2D tessellations such as the Voronoi-diagram and isohedral tessellations. The results show that this approach is feasible; *however, it is uncertain whether the approach is applicable to fairly complex three-dimensional forms, to other product typologies and to other manufacturing technologies as well.*

In this paper, we therefore examine the extent of the applicability of the approach by investigating algorithms originating in nature that allow for adapting complex three-dimensional forms to functional, manufacturing, and aesthetic constraints and objectives. It was also important to demonstrate that the output of nature-based computational means of form generation does not need to be confined to rapid prototyping, but can also be *realized with established fabrication technologies allowing for mass production* - whether using high or low tech materials. The forms are complex in the sense that creating them manually would be very time consuming and difficult; the constraints and objectives to which the forms must adhere further adding to the complexity of generating feasible forms.

In this paper, we apply the approach from [19] and [20] for two three-dimensional forms. We examined 1) Lindenmayer systems (L-systems) coupled to a genetic algorithm (GA) to create user-controlled branching support structures, and 2) minimal surfaces to create user-controlled lighting diffusers. We chose these morphologies based on their functional and aesthetic properties, and to represent two vastly different adaptation processes. Both these applications have a set of constraints and objectives associated with them in terms of functionality and manufacturability, described in Section 3. To verify this process a set of lights based on the minimal surfaces have been built and have been selected for exhibition at several international design fairs (DMY 2011, Stockholm Furniture Fair 2013, Biennale Internationale Design Saint-Etienne 2013) showing that the generative design system made it possible to produce a fully developed, "ready-for-sale" product, with potential for large-scale production. These studies reinforce the feasibility of using both forms derived from nature and their generation methods for product design.

2. Related works

Complex natural-mathematic morphologies have been the subject of interest in the artistic world for quite some time with artists such as Herbert W. Franke and Peter Henne creating algorithm-based computer graphics already in the 1950s and early 1960s. Within computer based bio-inspired generation of art and music, works by Sims [27], Todd and Latham [29] and Romero [24] show the wealth of research conducted on the topic.

In architecture, Lynn [16], Kolarevic [12] and Oxman [21] represent important works showing the extent to which digital design tools and generative design have been adopted by the architectural community.

Some works in design computing are taking into account functional or technical constraints and aesthetics. Shea and Cagan [26] use a combination of shape grammar and simulated annealing for both functional and aesthetic purposes and applied it for truss structures. Their model is re-used in [14] (shape grammar and GA) to develop stylistically consistent forms applied to the design of a camera. The designs generated took into account the constraints linked to the spatial component configuration. A designer was in charge of the aesthetic evaluation, following the interactive genetic algorithm paradigm [11]. Common are also the use of evolutionary methods to optimize a parameterized geometry in relation to objectives such as minimize weight or structural rigidity (see e.g. [1], [2]).

Examples of natural-mathematical morphologies in product design can mostly be found in industry with examples such as Trubridge's polyhedral ceiling lamp [30] and Wertel and Oberfell's Fractal-T [31] table. Many of the products based on complex morphologies are, however, produced by rapid prototyping, many examples can be found at rapid prototyping providers such as Shapeways (www.shapeways.com/) where consumer can also customize products such as jewelry developed by for instance Nervous System (http://n-e-r-v-o-u-s.com/). Products realized with traditional materials include Kram/Weisshaar’s Breeding Tables project, which generates variations of a table design using a GA that modifies a set of parameters ruling the support structure [13]. The system does not take stability into account, but it does ensure the producibility of the designs through constraints on the parameters. The Computational Chair project developed by EZCT Architecture & Design research [6] also uses a GA to generate design variations of a chair built from pieces of plywood glued together, but the algorithm in this case also minimizes the weight and ensures the structural stability of the chair through finite-element analysis. These examples show that although complex morphologies have been used in design, they are most often items which have no or few constraints such as structural stability, and are made by the process of rapid prototyping which permits almost arbitrary shapes to be produced, but is so far prohibitively costly for the production of larger structures such as tables or chairs, and unless specialty techniques and materials are used lack the surface quality and structural strength required for useful objects. Examples such as Kram/Weisshaar's tables and EZCT Architecture & Design research's chairs are built with traditional materials and do take into consideration the structural rigidity of the product, but are not mass produced objects, but rather one-off art-designs.

In this work, we therefore focus on generating designs realized with established fabrication technologies allowing for mass production based both on the use of natural forms and on the use of the adaptive natural processes for taking into account manufacturing constraints, functionality and aesthetic properties.

3. Implementation

3.1. Support structure with L- systems and GA

L-systems

An L-system, is a shape grammar ruling how a structure grows, used for example to model the growth of plants and some organisms [15], [23] (see Figure 1 for examples of structures in nature that are typically modeled with L-systems). L-systems have been used for artistic purposes and linked to GAs in for instance [17] and [9]. L-systems can also be used to generate self-similar fractals. L-system-based two- or three-dimensional structures have an irregular branch-like formal aesthetic and connect points on the plane or in space from a central node. An L-system is defined by a set of variables or sub-segments that can be used in the structure, a starting point, and a set of rules describing in what way the sub-segments can be combined. Their infinite variability makes them suitable for highly individualized yet self-similar objects. L-systems are useful for the generation of structures reaching to points in space or target points on a space envelope or for the generation of spatial lattices assembled from a possibly limited number of discrete elements. Because of their irregularity, objects based on L-systems are preferably manufactured via additive fabrication, laser/water jet cutting with subsequent computer numerically controlled (CNC) bending, and to a certain degree CNC milling.

Even with relatively few possible angles and lengths, an L-system can generate millions of possible branching structures, which in itself is not a problem until functional and manufacturing constraints and objectives come into the picture. Given objectives and constraints it becomes apparent that a human designer would require many years to sift through and evaluate the possible branching structures stemming from the L-system definition. What is needed is an optimization algorithm that does this automatically and quickly while leaving the user to evaluate the qualitative aspects - such as aesthetics - of the few solutions satisfying the constraints. There are numerous algorithms for optimizing structures; one of the most frequently used algorithms for non-linear modular structures is the GA [7]. A GA is a search heuristic that mimics the process of natural evolution. A GA treats each candidate solution as an individual, encoded by its genotype. Together, the individuals create a population of candidate solutions. The individuals in the population are then, depending on their fitness in relation to the constraints and objectives, modified by processes such as inheritance, mutation, selection, and crossover to create the next generation of the population.

Figure 1. L- systems in nature: Mangrove roots[1], Redwood branches[2], Fern leaf[3]

Application

An L-system is ideal for creating structures from a limited set of elements, adhering to some rules. Therefore, an L-system was applied to create a branching structure composed of individual pieces manufactured by laser cutting and CNC-bending sheet metal (see Figure 2). In this application the goal was to create a structure that would connect a point in space to a plane, similarly to a support structure for a roof. To limit the number of types of pieces needed to be manufactured, the pieces used should be discrete, meaning their length and angles should have a limited number of possible values. In this application the constraints were: no intersection of the branches, no intersection between the branches and the support surface and a certain required number of branch ends; the objective was that the ends of the branches should be as close as possible to the support plane.

Implementation

There are many applications available for generating L-systems, such as Branching [18], powerPlant [25] or L-studio [10]. However, few support export to common 3D file-formats, and none can be scripted to evaluate the properties of the L-system. Therefore a custom L-system generating script was implemented in Matlab to gain full control of the structure. Matlab has a wide array of optimization tools built in, which makes the connection of the L-system generation to the GA efficient. The L-system used for this application consists of a fixed starting point, a number of branches with different lengths and angles, which in turn have a number of branches connected to their ends, and so on. The L-system script takes as input the requested number of branching levels in the structure, the maximum number of new branches at the end of each branch, the branch lengths allowed, and the branching angles allowed. The script takes this input and creates random branching structures from the data by combining different branch lengths and angles.

A GA was used to find satisfactory branching structures. The GA used is the standard Matlab implementation with rank as scaling method, stochastic uniform as selection method, Gaussian as mutation function, single point as crossover function, elite count 2, and crossover fraction 0.8. The GA was run with a population of 150 individuals, during a maximum of 1500 generations. A GA represents an individual as a genotype. In this case the

---

[1] © 2003 Cesar Paes Barreto, available at http://www.sxc.hu/photo/40192

[2] © 2011 Lukas Osinski / CC BY-SA 3.0, available at commons.wikimedia.org/wiki/File:Metasequoia_glyptostroboides_Marki_branches.jpg

[3] © 2008 Forest and Kim Starr / CC BY-SA 3.0, available at commons.wikimedia.org/wiki/File:Starr_080117-2229_Microsorum_musifolium.jpg

genotype consisted of one instance of the L-system, i.e. a list of building instructions such as "add a branch with length 100 and angle 30° to branch 2". This genome is then interpreted by the GA to create a phenotype, in this case the 3D geometry which was evaluated by the system. The GA could mutate the branching structures by shifting their branch lengths and angles between the predefined values. The crossover has been done by grafting random parts of two branching structure parents into one child structure. The evaluation function scores each structure in accordance to how well it fulfills the constraints and objective. Once the optimization has ended, the resulting solutions satisfying the constraints and minimizing the objective can be visualized on screen, reviewed by the industrial designer, and sent to a surface modeling software where drawings of the parts could be created and used for fabrication (see Figure 3).

Figure 2. Structure mock-up of one of the generated L-systems

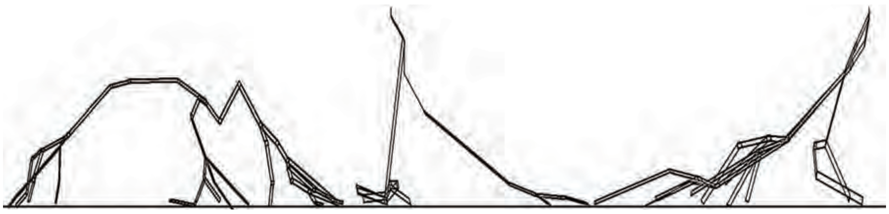


Figure 3. Output of the optimization process showing three different solutions that satisfy the constraints

3.2. Controlled lighting diffusers with minimal surfaces

Minimal surfaces

Minimal surfaces were originally a name for surfaces that minimized surface area, subject to some constraint, such as total volume enclosed or a specified boundary, with research on the subject dating back to the 18th century (see e.g. [8, pp. 1-5]), but the term is now used more generally to describe a surface with a mean curvature of zero [4]. Minimal surface based three-dimensional structures have self-similar tent-like formal aesthetic. They are spanning connected boundaries in space. Their infinite variability makes them suitable for highly individualized yet self-similar objects. Minimal surfaces are useful for: the generation of lightweight load-bearing tensile structures, the finding of area and weight minimizing surfaces within given boundaries. Architects such as Frei Otto and Barry Patten and artists such as Robert Engman and Robert Longhurst have used these properties of minimal surfaces before computational methods became prevalent. More recently, projects such as that of Design Research Exchange[5] show how modern software for form generation and structural evaluation can be combined for high-rise structures. Because of their irregularity, objects based on minimal surfaces are preferably manufactured via additive fabrication, CNC milling, vacuum forming and to a certain degree via concrete casting, metal casting, metal stamping, slip casting.

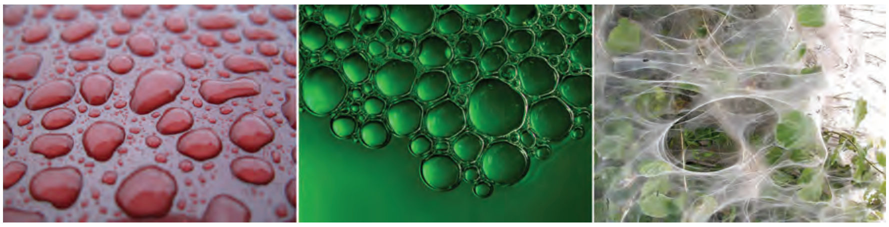


Figure 4. Minimal surfaces in nature: raindrops[4], foam[5], caterpillar webs[6]

Application

In order to utilize the properties of the minimal surface, surface area minimization, while keeping the volume constant, slip cast porcelain shapes were deemed suitable. Because of porcelain's translucency and its matte surface when unglazed, the shapes were used for reflecting and diffusing light from high powered LEDs. The surface minimization yields surfaces which are optimal in terms of material usage for containing a certain volume. We wanted to use surface minimization to simulate a drop of water with a user-defined bottom contour resting on a flat surface. In order for a user of the application to be able to control the shape, the 2D contour of the initial shape should be possible to modify (see Figure 5), and the resulting shape of the surface minimization should be easy to review. The objective is thus to minimize the surface, while constraining the contour and volume of the shape.

Implementation

To generate and manipulate a minimal surface there are a few alternatives, the analog way that architects such as Frei Otto and Barry Patten used consists in using materials that seek to minimize their surface energy, and thereby generating minimal surfaces by their physical properties such as elastic fabrics, liquids, and soap films that are constrained by boundary wireframes. However, this method is time-consuming and difficult to control,

---

[4] © 2007 Andrew Bossi / CC BY-SA 2.5, available at commons.wikimedia.org/wiki/File:2007_10_25_-_Greenbelt_ _Water_drops_on_a_Saab_9-3_roof_2.JPG

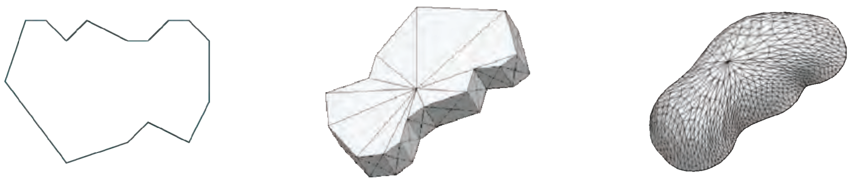
[5] © 2007 woodleywonderworks / CC BY-SA 3.0, available at www.flickr.com/photos/wwworks/667298782/

[6] © 2006 Penny Mayes / CC BY-SA 2.0, available at commons.wikimedia.org/wiki/File:Abstract_art_in_the_hedgerow_-_geograph.org.uk_-_178953.jpg

and does not easily translate to technical manufacturing instructions. A more efficient and versatile method is to generate the surfaces digitally through software written to simulate the physics of surface tension and gravitation.

Many programs for generating minimal surfaces exists, such as Ken Bracke's Surface Evolver (SE) [3] that is versatile and powerful. However, it requires that the input to the application is scripted in a specific language. In order to be able to easily control the contour and height of the form, without the user having to hard-code geometry into a SE-script, a custom written script in Matlab was created which takes as input a contour of a 2.5D volume (see Figure 5a), and outputs a script directly to SE. In this application the code tells SE to treat the input geometry as a volume of water resting on a surface under the influence of gravitation and a wetting angle between the liquid and the surface. The geometry resulting from SE's minimization of surface energy can be displayed on-screen for evaluation by the designer (see Figure 5c). It enables the user to sketch and modify the contour and height of the shape, and then get instant feedback of the resulting minimal surface. If the surface is deemed interesting it can be exported for use with all major surface modeling software such as Rhino or Alias. Using a surface modeling software, thickness can be added to the surface (see Figure 6a) and then the thickened shell can be used as input to 3D-printing software, or computer-aided manufacturing (CAM) software such as ArtCam, which are then able to generate instructions for a CNC mill to cut the shape from a block of model material. The milled or 3D-printed model can be used as is (see Figure 6b), or used as master models for creating plaster molds for casting ceramics and plastics (see Figure 6c).

Three minimal surface master models were produced using this workflow, two were 3D-printed, and one was CNC-milled. The cost of 3D-printing and other similar rapid prototyping techniques is still somewhat prohibiting when outputting large shapes, and it is therefore the largest shape was milled. The durability of the 3D-printed models when in contact with moisture also make them less than ideal, however, it should be noted that the surface quality generated by rapid-prototyping is often good enough to use directly for mold making without any extra finishing needed, whereas the surfaces resulting from milling might require sanding and filling. The three shapes were then used as master models for making plaster molds, which in turn were used to cast ceramic shells that were later fired (see Figure 7).



(a) (b) (c)

Figure 5. a) User defined drop contour, b) initial extruded surface block, c) first surface energy minimization step

(a) (b) (c)

Figure 6. a) Thickened mesh, b) 3D printed model with support ribbing, c) mold-making

Figure 7. Final pieces after firing

4. Discussion and conclusion

This paper has demonstrated that fairly complex everyday objects based on three-dimensional natural-mathematical morphologies can be designed, evaluated and produced into a solid form; that digital and mass production methods can be linked to create an aesthetic and functional whole beyond purely decorative mimicry. We showed that the approach is not limited to two-dimensional morphologies and 2.5D objects by describing a software-based process to design with three-dimensional morphologies which was used to generate porcelain diffusers for high-power LED lighting, and stainless steel support structures (pending realization), thus further validating the approach proposed in [19] and [20]. In the case of the porcelain diffuser, the output from the generative design system made it possible to produce a fully developed, "ready-for-sale" product, with potential for large-scale production. This is a step towards enabling industrial designers the same level of form articulation as has been available to artists and architects, even though the constraints on the design activity are much different. The user could easily change the inputs to the algorithm, either through text-based or graphical interaction, and get feedback of the resulting forms. For the minimal surface application, the feedback of the optimized shape was almost instantaneous, generally requiring less than a second for returning an optimized shape. The optimization process for the L-system required more time, usually around 3 minutes to converge on a satisfactory solution. The design tools could find solutions that satisfied the constraints. For the minimal surfaces, the intended and final volumes were identical. The structures generated by the L-system and GA all complied with the constraint that no parts should intersect. In terms of optimization the surface minimization gave material savings ranging from 31%-23% between the original form defined by the user, and the resulting SE output, while keeping the volume constant, while the L-system optimization could find structures that were on average 90% closer to the support surface than the starting L-system.

Access to an enhanced morphological repertoire, exploiting fully the possibilities to design with nature-derived forms, and, resultantly, enhanced creativity could benefit industrial designers. Other benefits can be envisioned: the emergence of digital crafts might enable relocating production to the vicinity of consumption, and rededication of existing production methods and equipment to produce individualized products could become a reality. In the extension of our approach, introducing generative design tools to consumers might lead to participatory and community-based designing, on- or offline, linked to digital fabrication.

Realizations from industrial designers often take place in the industrial context of product development. Compared to a regular product development process, it is clear that an approach such as that described in this paper can be beneficial for creating products that are individualized for every application. However, given the extra effort required to implement a generative design system for a chosen morphology and product, it might not be economically feasible for mass production. In this paper, two algorithms for form generation found in nature have been demonstrated, but the approach could be used with morphologies from other origins. Future topics of research include how applicable the approach is to more complex products such as dynamic systems.

Acknowledgements

This work was supported by Innovativ Kultur that funded the project "Putting Nature to Work". The authors would also like to thank Assistant Professor Jenny Janhager Stier at the department of machine design, KTH who was the project leader.

References

[1] Bentley, P. J., 1999, *Evolutionary design by computers*, Morgan Kaufmann, San Francisco, CA.

[2] Bentley, P. J. and Corne, D. W., 2002, *Creative Evolutionary Systems*, Morgan Kaufmann, San Francisco, CA.

[3] Brakke, K. *Surface Evolver*. Retrieved from: http://www.susqu.edu/brakke/evolver/evolver.html, last accessed: 6-14-2012.

[4] Dierkes, U., Hildebrandt, S. and Sauvigny, F., 2010, *Minimal Surfaces*, Springer Berlin Heidelberg, Berlin, Heidelberg.

[5] DRX Design Research Exchange, 2012, *Minimal Surface High-Rise Structures*.

[6] EZCT Architecture & Design Research, Hamda, H. and Schoenauer, M., 2004, *Studies on Optimization: Computational Chair Design using Genetic Algorithms*, EZCT Architecture & Design Research, Paris.

[7] Goldberg, D. E., 1989, *Genetic Algorithms in Search, Optimization, and Machine Learning*, Addison-Wesley, Reading, MA.

[8] Isenberg, C., 1992, *The Science of Soap Films and Soap Bubbles*, 2nd Edition, Dover, New York, NY.

[9] Jacob, C., 1994, "Genetic L-System Programming", in Davidor Y., Schwefel H.-P. and Männer R. (Eds), *Parallel Problem Solving from Nature — PPSN III*, 1994, Vol. 866, Springer, Berlin Heidelberg, pp. 333-343.

[10] Karwowski, R. and Lane, B. *L-studio*. Retrieved from: http://algorithmicbotany.org/lstudio/, last accessed: 6-1-2004.

[11] Kelly, J. C., 2008, *Interactive Genetic Algorithms for Shape Preference Assessment in Engineering Design*, ProQuest, Ann Arbor, MI.

[12] Kolarevic, B., 2003, *Architecture in the digital age: design and manufacturing*, Spon, London.

[13] Kram, R. and Weisshaar, C., 2003, *Breeding Tables*, <http://www.kramweisshaar.com/projects/breeding-tables.html>, last accessed: 22 February 2012

[14] Lee, H. C. and Tang, M. X., 2009, *Evolving product form designs using parametric shape grammars integrated with genetic programming*, Artificial Intelligence for Engineering Design, Analysis and Manufacturing - AI EDAM, 23(2), pp. 131-158.

[15] Lindenmayer, A., 1968, *Mathematical Models for Cellular Interactions in Development, Parts I*

and II, Journal of theoretical biology, 18, pp. 280-315.

[16] Lynn, G. and Papadakis, A. C., 1993, *Folding in architecture*, Academy, London.

[17] McCormack, J., 2004, "Aesthetic Evolution of L-Systems Revisited", in Raidl G. R., Cagnnoni S., Branke J., Corne D. W., Drechsler R., Jin Y., Johnson C. G., Machado P., Marchiori E., Rothlauf F., Smith G. D. and Squillero G. (Eds), *Applications of Evolutionary Computing*, 2004, Vol. 3005, Springer, Berlin Heidelberg, pp. 477-488.

[18] Mizuno, R. *Branching: L-system Tree*. Retrieved from: http://www.mizuno.org/applet/branching/, last accessed: 1-1-2006.

[19] Nordin, A., Hopf, A., Motte, D., Bjärnemo, R. and Eckhardt, C.-C., 2011, *Using genetic algorithms and Voronoi diagrams in product design*, Journal of Computing and Information Science in Engineering, 11(011006).

[20] Nordin, A., Motte, D., Hopf, A., Bjärnemo, R. and Eckhardt, C.-C., 2010, *Complex product form generation in industrial design: A bookshelf based on Voronoi diagrams*, DCC'10,Springer, pp. 701-720.

[21] Oxman, R., 2006, *Theory and design in the first digital age*, Design Studies, 27(3), pp. 229-265.

[22] Pearce, P., 1978, *Structure in Nature is a Strategy for Design*, MIT Press, Cambridge, MA.

[23] Prusinkiewicz, P., Hanan, J. S. and Lindenmayer, A., 1990, *The algorithmic beauty of plants*, Springer-Vlg, Berlin .

[24] Romero, J. and Machado, P., 11-22-2007, *The Art of Artificial Evolution: A Handbook on Evolutionary Art and Music*, Springer, Berlin Heidelberg.

[25] Rosanwo, O. and Gleske, P. *powerPlant*. Retrieved from: http://sourceforge.net/projects/pplant/, last accessed: 9-28-2009.

[26] Shea, K. and Cagan, J., 1999, *Languages and semantics of grammatical discrete structures*, Artificial Intelligence for Engineering Design, Analysis and Manufacturing - AI EDAM, 13(4), pp. 241-251.

[27] Sims, K., 7-4-1991, *Artificial evolution for computer graphics*, SIGGRAPH '91, Vol. Volume 25 Issue 4,Association for Computing Machinery, pp. 319-328.

[28] Thompson, D. W., 1945, *On Growth and Form*, 2nd Edition, Cambridge University Press, Cambridge, MA.

[29] Todd, S. J. P. and Latham, W., 1992, *Evolutionary art and computers*, Academic Press Inc, London and San Diego, CA.

[30] Trubridge, D., 2010, *Coral [lamp]*, <http://www.davidtrubridge.com/coral/>, last accessed: 29 January 2011

[31] Wertel, J. and Oberfell, G., 2007, *Fractal-T* [Table]. In Klanten, R., Ehmann, S., Kupetz, A., Moreno, S., and Mollard, A. (Eds), *Desire - The Shape of Things to come*, Die Gestalten Verlag, Berlin, 2008, p. 134.

Paper VI

Constraint-handling techniques for generative product design systems in the mass customization context

VI

Nordin, A., Motte, D., Hopf, A., Bjärnemo, R., & Eckhardt, C.-C.

Artificial Intelligence for Engineering Design, Analysis and Manufacturing

Originally published by Cambridge University Press

Artificial Intelligence for Engineering Design, Analysis and Manufacturing (2013), *27*, 387–399.

doi:10.1017/S0890060413000383

Constraint-handling techniques for generative product design systems in the mass customization context

AXEL NORDIN, DAMIEN MOTTE, ANDREAS HOPF, ROBERT BJÄRNEMO, AND CLAUS-CHRISTIAN ECKHARDT
Department of Design Sciences, Faculty of Engineering LTH, Lund University, Lund, Sweden

(RECEIVED June 15, 2012; ACCEPTED May 1, 2013)

Abstract

Generative product design systems used in the context of mass customization are required to generate diverse solutions quickly and reliably without necessitating modification or tuning during use. When such systems are employed to allow for the mass customization of *product form*, they must be able to handle mass production and engineering constraints that can be time-consuming to evaluate and difficult to fulfill. These issues are related to how the constraints are handled in the generative design system. This article evaluates two promising sequential constraint-handling techniques and the often used weighted sum technique with regard to convergence time, convergence rate, and diversity of the design solutions. The application used for this purpose was a design system aimed at generating a table with an advanced form: a Voronoi diagram based structure. The design problem was constrained in terms of production as well as stability, requiring a time-consuming finite element evaluation. Regarding convergence time and rate, one of the sequential constraint-handling techniques performed significantly better than the weighted sum technique. Nevertheless, the weighted sum technique presented respectable results and therefore remains a relevant technique. Regarding diversity, none of the techniques could generate diverse solutions in a single search run. In contrast, the solutions from different searches were always diverse. Solution diversity is thus gained at the cost of more runs, but no evaluation of the diversity of the solutions is needed. This result is important, because a diversity evaluation function would otherwise have to be developed for every new type of design. Efficient handling of complex constraints is an important step toward mass customization of nontrivial product forms.

Keywords: Complex Morphologies; Constraint-Handling Techniques; Evolutionary Computing; Generative Design; Genetic Algorithms

1. INTRODUCTION

One can sense an evolution of the largely static relationship between the consumer and the product. There is an increasing desire to participate in the designing of products and the potential experiences consumers will share with them. As put forward by Friebe and Ramge (2008), the upsurge of independent fashion labels, "crowdsourcing" initiatives, and coworking spaces indicates consumer's demand for empowerment. This need for cocreation, implemented already in the textile (Lakhani & Kanji, 2009) and food industries (Kraft Foods, 2006) but also in more advanced consumer goods businesses like sportswear (Moser et al., 2006; Bouché, 2009), goes well beyond mere material and color choices: the future "prosumer" (Toffler, 1971) desires control over product form as well. This challenge poses major difficulties. First, consumers are not always knowledgable enough to evaluate how their design preferences may affect the functionality and manufacturability of the product. Second, if the desired product form is complex, such as in nature-inspired forms or shape grammars, even the manipulation of the form can be too cumbersome for consumers not skilled in three-dimensional modeling. The consumer must therefore be supported in form manipulation in some way. Finally, if mass customization is understood as the mass production of customized goods (Kaplan & Haenlein, 2006; Trubridge, 2010, p. 169), the product form is often severely constrained by the production system.

A possible solution to these difficulties is to implement a *generative design system* (GDS) that generates product designs that fulfill mass production and engineering constraints, along with consumer requirements (such as size, contour, and materials), while leaving the consumer in control of the final

Reprint requests to: Axel Nordin, Division of Machine Design, Department of Design Sciences, Faculty of Engineering LTH, Lund University, P.O. Box 118, 221 00 Lund, Sweden. E-mail: axel.nordin@mkon.lth.se

design selection. A GDS intended for product design is basically structured around a graphical user interface with which the user can evaluate, select and influence the generation of product forms. A GDS is often based on an interactive optimization system or constraint satisfaction system that handles user preferences and technical constraints. A GDS is frequently used to handle complex forms usually intractable to the user. Most GDSs have been intended for professional designers, for example, to help the designer preserve the "form identity" of a brand (Pugliese & Cagan, 2002; Chau et al., 2004; McCormack et al., 2004), but they have not been specifically designed for use by consumers. Letting the consumers control their own design adds a number of requirements to the GDS.

First, such a GDS is to be used repeatedly; therefore, the solutions must be generated *quickly* (how fast the system is able to converge to viable solutions, i.e., *convergence time*, is important) and in a reliable manner (how often the system is able to converge to viable solutions, i.e., *convergence rate*, is important), and the system must be applicable to a wide range of problems without requiring extensive modification of the algorithm by the consumer or a programmer.

Second, consumers must be able to choose from a set of solutions, because the decision to choose one design solution over another is often not based on pure performance metrics but rather on criteria that are subjective and difficult to quantify. At the same time, in order to give the consumer a meaningful choice, the generated shapes need to fulfill all technical constraints, which may be time-consuming to evaluate and hard to satisfy. For an analysis related to structural problems, for instance, finite element techniques may be required. It is therefore necessary that the GDS ensure an adequate *diversity* among the proposed solutions so that the waiting time and the need to relaunch the generation process are minimized.

Diversity, convergence rate, and convergence time are interrelated: they depend upon how the solutions are generated, that is, how the constraints are handled and the viable solutions are optimized. Of these two activities, the satisfaction of constraints represents the main challenge. The time spent on the optimization can be controlled by the user (optimization can be stopped if deemed to be too time-consuming or if one is satisfied with the result), but the constraint-handling step cannot. Regarding diversity, the constraints can be very hard to satisfy, and the space of feasible solutions in those cases is sparse. Diversity is therefore unlikely to arise during the subsequent optimization step if it has not during the constraint-handling step. Finally, the convergence rate also depends on the constraint-handling step, because solutions are viable only if they fulfill all constraints. Therefore, in the following discussion, these issues are considered only under the constraint-handling aspect.

Enabling the efficient handling of such types of constraints is a step toward showing that mass customization of product forms is technically possible. In this paper, three different techniques for handling technical constraints are therefore evaluated in terms of convergence time, convergence rate, and the diversity of the generated solutions using a real design problem. The design problem is to find feasible solutions to a table support structure based on a complex form (a so-called Voronoi diagram) that is subject to technical constraints and user evaluation.

Section 2 reviews related works on GDSs and on constraint-handling techniques (CHTs). The study of diversity, convergence rate, and solution-generation time for a real design problem with selected CHTs is treated in Section 3.

Although this research addresses primarily the use of GDSs in the mass customization context, some aspects of it, especially that pertaining to the diversity issue, should also be useful for GDSs where the user is a professional industrial designer.

2. GDSs AND CHTs

The first part of this section reviews related works on GDSs and reports how constraints are handled in these systems. The second part reviews CHTs and their relevance for consumer-oriented GDSs.

2.1. Related works on GDSs

Few GDSs focus on product forms within the mass customization context. Current GDSs are mainly industrial applications in the form of online product configuration websites offering many diverse forms of mass customization, a large bandwidth of personalization options, navigation techniques, and visual quality. A collection of these websites can be found at MilkorSugar (http://www.milkorsugar.com/). One example is the Kram/Weisshaar Breeding Tables Project, which generates variations of a table design using a genetic algorithm (GA) that modifies a set of parameters ruling the support structure (Kram & Weisshaar, 2003). Despite the steady upsurge in online product configuration, there is no major market player that makes customization of the actual product form and structure available to its customers. It should also be noted that none of these configurators includes evaluation of manufacturing or structural constraints.

Within industrial design research, generative design has been investigated primarily for stylistic purposes. In the seminal work of Knight (1980), a parametric shape grammar was developed for the generation of Hepplewhite-style chairbacks. Orsborn et al. (2006) employed a shape grammar to define the boundaries between different automotive vehicle typologies. Recent works have focused on branding-related issues. With the help of shape grammars, new designs based on the Buick (McCormack et al., 2004), Harley-Davidson (Pugliese & Cagan, 2002), and Coca-Cola and Head & Shoulders (Chau et al., 2004) brands were developed. Further research is being undertaken to develop rules that link form and brand: for example, Cluzel et al. (2012) for systems based on GAs and Orsborn et al. (2008) for shape grammars. Within the mass customization context, Johnson (2012) created a graphical interface for customizing shelves while taking into account functional aspects such as compartment size, and Piasecki and Hanna (2010) used a graphical interface based on numerical sliders to control a shape, as well as a GA to aid in the design of the shape, to investigate the influence of the amount of control on the user's satisfaction with the system.

Technical constraints and objectives have been studied more extensively within engineering design. An early example is Frazer's application of a GA to the design of sailing yachts, taking into account constraints such as stability, center of buoyancy, and wetted surface area, as well as less well-defined criteria such as aesthetics, by combining a computational evaluation with a subjective user-based evaluation (Frazer, 1996). Agarwal and colleagues (Agarwal & Cagan, 1998; Agarwal et al., 1999) have associated the *shape* grammar technique with parametric cost and applied it to the design of coffee makers (see Cagan, 2001, for a review on the use of shape grammars in engineering design). Numerous efforts have also been made to take into account the engineering and production constraints early in the development process, using knowledge-based engineering systems (see El-Sayed & Chassapis, 2005; Sandberg & Larsson, 2006; Lin et al., 2009; Johansson, 2011) or a combination of knowledge-based engineering and optimization systems, as in Petersson et al. (2012), where the lightweight gripper constraint satisfaction and optimization system is based on the weighted sum technique. These works, even if they present interesting design systems, are not primarily concerned with diversity and choice.

Some works are crossing the boundaries between engineering and industrial design, taking into account functional or technical constraints and aesthetics. Shea and Cagan (1999) used a combination of shape grammar and simulated annealing for both functional and aesthetic purposes and applied it to truss structures. The shape grammar technique was used to generate new designs, and the simulated annealing technique to direct the generation toward an optimum. The evaluation was based on a weighted sum of constraint violations and objective values. The design objectives were functional (minimize weight, enclosure space, and surface area), economic, and aesthetic (minimize variations between lengths in order to get uniformity, make the proportions respect the golden ratio). Shea and Cagan's model was reused by Lee and Tang (2009), with a combination of shape grammar and GA, to develop stylistically consistent forms and it was applied to the designing of a camera. The designs generated took into account the constraints linked to the spatial component configuration. The constraints were handled by minimizing a weighted sum of the constraint violations. A designer was in charge of the aesthetic evaluation, following the interactive GA paradigm. Ang et al. (2006) used shape grammars and GAs to develop the Coca-Cola bottle example of Chau et al. (2004) and added functional considerations (the volume of the bottle) that were constrained to approach the classic Coca-Cola bottle shape. EZCT Architecture & Design Research et al. (2004), within the interactive GA paradigm, developed a set of chairs optimized for weight and stiffness. The designer could define how loads would be applied to the structure before the optimization but could not interact with the system during the optimization. Finally, Wenli (2008) developed a system that, through adaptive mechanisms, allowed it to learn the designer's intent faster; that system was implemented as a plug-in for a computer-aided design system and applied to boat hull design.

The handling of the constraints in the reviewed works is summarized in Table 1. Of the CHTs reviewed in the previous section, the weighted sum technique is always used if more than one constraint or objective is present.

2.2. CHTs

CHTs represent a field of evolutionary computing that is increasing at a fast pace. There are several techniques (for extended reviews, see Michalewicz et al., 1996; Coello Coello, 2002; Mezura-Montes, 2004; Yeniay, 2005). As mentioned in the Introduction, GDSs for mass customization will be used repeatedly. It is therefore necessary to have CHTs that are sufficiently generic for addressing different design problems and that do not require the user to modify the algorithm during use. Many of the common types of CHTs are therefore not applicable, as discussed below.

The most common approach to handling constraints is to use methods based on penalty functions. The concept behind those methods is "to transform a constrained-optimization problem into an unconstrained one by adding (or subtracting) a certain value to/from the objective function based on the amount of constraint violation present in a certain solution" (Coello Coello, 2002). The penalty factors/values must be determined by the user and is problem dependent (Mezura-Montes & Coello Coello, 2006, p. 2). The weighted sum can be seen as one specific penalty technique: the constraints are incorporated into the objective function and the given weights that penalize the fitness value. Another type of CHT consists of trying to maintain feasibility of the solutions (Michalewicz & Janikow, 1991; Schoenauer & Michalewicz, 1996); it requires a feasible starting point that may be computationally costly to find or that must be set by the user (Coello Coello, 2002, p. 1259) and/or necessitates the use of problem-specific operators (Schoenauer & Michalewicz, 1996, p. 245). Another method is based on the search for feasible solutions. One possibility is "repairing" infeasible individuals (see details in Coello Coello, 2002, section 4), which has been proved an efficient method if the individuals can be easily transformed; this unfortunately is rarely the case in real-world engineering problems. Hybrid methods also exist that combine techniques from the different categories above and/or with techniques from other domains, such as fuzzy logic (Van Le, 1996) or constraint satisfaction problems (Paredis, 1994; see also Michalewicz & Schoenauer, 1996; and Coello Coello, 2002). They require supplementary knowledge from the user for their implementation; they have therefore not been investigated further.

Table 1. *Comparison of the constraint handling of the reviewed works*

Work	KBS	Optimiz. System	CHT Type	Constraints	Objectives	Diversity Meas.	Time-Consum. Eval.
Knight, 1980	No	—	—	NA	NA	No	No
Orsborn et al., 2006	No	—	—	NA	NA	No	No
McCormack et al., 2004	No	—	—	NA	NA	No	No
Pugliese & Cagan, 2002	No	—	—	NA	NA	No	No
Cluzel et al., 2012	No	GA	—	NA	NA	Yes[a]	No
Orsborn et al., 2008	No	—	—	NA	NA	No	No
Agarwal & Cagan, 1998	No	—	—	NA	NA	No	No
Lin et al., 2009	Yes	—	—	NA	NA	No	No
Johansson, 2011	Yes	—	—	NA	NA	No	No
Sandberg & Larsson, 2006	Yes	—	—	NA	NA	No	No
El-Sayed & Chassapis, 2005	Yes	ND	ND	3	2	No	Yes
Petersson et al., 2012	Yes	SA	WS	1	1	No	Yes
Shea & Cagan, 1999	No	SA	WS	4	5	No	Yes
Lee & Tang, 2009	No	GA	WS	1	3	No	No
Ang et al., 2006	No	GA	—	1	0	No	No
Chau et al., 2004	No	—	—	NA	NA	No	No
EZCT Architecture & Design Research et al., 2004	No	GA	WS	1	1	No	Yes
Wenli, 2008	No	GA	—	0	NA	No	No

Note: The number of constraints listed are those which are handled by the constraint-handling technique (CHT). Constraints handled by knowledge base systems were not taken into account. In the case of interactive genetic algorithms (GAs), the objectives handled by users were not included. NA, not applicable; ND, no data; SA, simulated annealing; WS, weighted sum.

[a]Cluzel et al. (2012) develops a similarity measure to test the performance of their interactive GA, but it is not used in the generative design system itself.

The types of CHTs that seems to fit the above-mentioned requirements are the lexicographic, or sequential, CHTs (SCHTs) and the multiobjective optimization techniques. Coming from the domain of multicriteria decision models (e.g., see Bouyssou et al., 2006, pp. 188–191), the lexicographic method consists in considering each constraint separately, in a specific order. When the first constraint is fulfilled, the next constraint is considered. When all constraints are fulfilled, the objective function is optimized. Although these methods do not require extensive tuning of parameters while they are running, it is necessary to select in advance a sequencing of all constraints. However, this choice of sequence needs to be done only once, *before* the GDS will be used. This aspect is crucial, because the ordering of constraints significantly influences the results in terms of running time and precision (Michalewicz & Schoenauer, 1996). How to choose an optimal sequence has been described elsewhere (Motte et al., 2011). The multiobjective optimization techniques are based on transforming the constraints into objectives to fulfill. This is also a promising technique for engineering optimization problems (for reviews, see Coello Coello, 2002; Mezura-Montes, 2004; Mezura-Montes & Coello Coello, 2006),

In a preceding study (Motte et al., 2011), two SCHTs were investigated against the classic weighted sum scheme using the well-known 10-bar truss benchmark problem (Haug & Arora, 1979): the behavioral memory (BM) method (Schoenauer & Xanthakis, 1993; Michalewicz et al., 1996) and the SCHT (Lexcoht; Nordin et al., 2011). Regarding convergence time, Lexcoht was more often superior to BM, and both were far superior to the weighted sum technique. It is interesting that no significant differences between different weighting schemes were found: the unweighted sum scheme (UWS), a linearly weighted scheme, and an exponentially weighted sum scheme were tested. Finding a relevant weighting scheme therefore does not seem crucial for an efficient use of the weighted sum technique, making it a potentially interesting generic CHT.

SCHTs and the weighted sum technique are both therefore candidates for consumer-oriented GDSs. Although less efficient than Lexcoht in terms of convergence time, BM is interesting because its structure (presented below) is based on a diversity measure in order to allow for a greater diversity in solutions.

Therefore, it was decided to compare these three CHTs in terms of diversity, convergence rate, and convergence time. The weighted sum technique is implemented using the UWS. The implementations of Lexcoht and BM are presented in the following sections.

2.2.1. *Lexcoht*

Lexcoht (Nordin et al., 2011) can be described for each constraint by performing the following:

- Evaluate the constraint violation.
- If the constraint is satisfied: evaluate the next constraint.
- If the constraint is not satisfied: stop the evaluation and score the individual according to

$$p = \frac{m + (1 - a)}{c}, \tag{1}$$

where p is the individual's score, m is the number of constraints the individual satisfied up until the last constraint evaluated, a is the constraint violation of the last evaluated constraint, and c is the total number of constraints. Constraint violation a is normalized (e.g., a = minimal allowed value/observed value), which means that p also ranges from 0 to 1.

As a result of Eq. (1), an individual satisfying m constraints is certain to get a higher score than an individual satisfying $m - k$ constraints, $k \in [1, m]$. The score p is then used as fitness in the GA (see Section 3.2).

2.2.2. The BM technique

Schoenauer and Xanthakis (1993) describe another sequential approach: the BM technique. It is based on the BM paradigm (de Garis, 1990), in which several techniques have been implemented to increase the diversity of the population to avoid premature convergence around certain constraints. The algorithm is summarized below.

A randomly initialized population is optimized in regard to the first constraint. This continues until a certain percentage, the flip-threshold φ of the population, satisfies the constraint. The population is then optimized in regard to the next constraint, until φ percent satisfies the second constraint. Any individual not satisfying the prior constraints is given the score zero. This process continues until all constraints have been satisfied.

To maintain population diversity, a sharing scheme is used as described in Holland (1975) and Goldberg and Richardson (1987). This method reduces the fitness of individuals that are similar to each other to promote diversity. The user-defined parameter-sharing factor σ_{sh} is used to decide whether two individuals are similar; it is also used to calculate the sharing score sh_i, which is used to penalize individuals that are similar (described below). The score p for each individual can be described as $p = (M_t - C_i/sh_i)$, where C_i is the constraint violation and M_t is an arbitrarily large positive number equal to or greater than the largest constraint violation.

Furthermore, a restricted mating scheme as described by Deb and Goldberg (1989) is used which promotes mating of similar individuals to create fitter offspring. The parameter σ_{sh} is also used here to decide whether two individuals are similar.

This method thus requires the user to determine the flip-threshold φ and the sharing factor σ_{sh}. However, recommendations for tuning the last two are given in Schoenauer and Xanthakis (1993): "the order of magnitude of σ_{sh} can be approximated from below using large φ and increasing σ_{sh} until the required percentage of feasible points cannot be reached anymore. Slightly decreasing σ_{sh} should then allow to find good values for both σ_{sh} and φ."

3. THE STUDY

3.1. Objectives of the study

The first objective is to compare the ability of the three CHTs to generate sufficient diversity among the proposed solutions. The second objective is to compare their convergence times. The third objective is to compare their relative convergence rates. In this study, the comparison is based on the table generation problem, presented next.

3.2. The table generation problem

The design problem is to generate Voronoi diagram based table structures based on that satisfy three production and structural constraints. A Voronoi diagram is a structure that is often found in light and strong structures in nature (Pearce, 1978; Beukers & van Hinte, 2005), such as the wing of a dragonfly or the structure of bone marrow. The manufacturing processes used are laser cutting and computer numerical control sheet-metal bending. The geometry of the bending machine limits the flange lengths of the cells to be manufactured to no shorter than 30 mm, which we call constraint l, and the bending angles a minimum of 35°, which we call constraint a. The structural requirements limit the maximum vertical displacement of any part of the table to 2.5 mm, which we call constraint f.

The design problem is described in depth in Nordin et al. (2011). A GDS based on this design problem would allow the consumer to determine the contour of the tabletop (see Fig. 1), to choose the height of the table, and to select the table's structure material. The GDS would then generate design proposals for the consumer to choose from. In this setup, the contour is chosen to be a square one. Note that prototypes have been built based on the computer-generated proposals and presented at several design fairs (see Fig. 2). This application can also be considered as a "real" design problem.

3.3. Implementation of the whole GDS

The table structure is represented as joined-beam elements, which are analyzed using the finite element method, using a finite element package called CALFEM developed at Lund University (Austrell et al., 2004). This package allows for defining a number of degrees of freedom for the cells, their positions and interconnections, as well as applicable loads and boundary conditions.

The GA used is the standard Matlab implementation. The scaling method used to assign probabilities for selection to the individuals is a simple ranking scheme where the individuals are ordered after their fitness; this approach avoids giving individuals with high fitness an unfair advantage in selection, which can result in premature convergence on local optima. The selection method chooses parents based on the individuals' scaled fitness, in this setup Matlab's built-in selection method *stochastic uniform* has been chosen. The stochastic uniform method represents the population as a line, with each individual representing a line segment whose length is proportional to the individual's scaled fitness. The method then walks down the line in fixed-length steps, adding the individual whose line segment it lands onto the pool of parents. The top two individuals are guaranteed to survive to the next generation in order to not lose the best so-

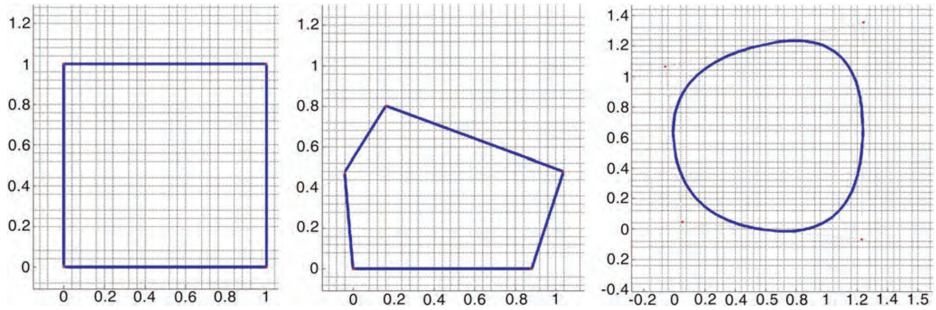


Fig. 1. (Color online) Three user-defined table contours.

lutions. The fraction of the children created by crossover, rather than mutation, is set to 0.8. The GA is run with a population of 50 individuals, during a maximum of 500 generations. The run is stopped when the maximum number of generations is reached or an individual satisfying all constraints has been found. Each original population was generated by randomly generating 70 Voronoi points for each of the individuals in the population.

Sharing score: Diversity measure of BM. The measure for diversity is based on the calculation of the sharing score for the BM method. The diversity of an individual in a population is calculated by comparing its genome, in this case the coordinates of its Voronoi points, to all the other genomes of the rest of the individuals in the population. This is achieved by the following pseudocode:

For each individual i in the population:
For each individual j in the population:
For each point a in individual i's genome:
Find the point b in individual j's genome that has the smallest Euclidian distance d_a to point a
The sum of all the distances $d_{i,j} = \sum_{a=1}^{70} d_a$ is individual i's diversity to individual j.

Fig. 2. (Color online) An image of the generated table (Nordin et al., 2011).

3.4. Experimental setup and procedure

Because there are only three constraints, all six possible sequences are investigated for the Lexcoht and BM techniques. In this paper, each sequence is named after the order in which the constraints are evaluated (*laf*, *lfa*, *alf*, *afl*, *fla*, and *fal*, respectively). The sequencing has no effect on the UWS, because all constraints are evaluated simultaneously. The investigation of the three CHTs therefore amounts to 13 "treatments" to investigate. The parameters for the BM techniques were set according to the recommendations from Schoenauer and Xanthakis (1993), with $\varphi = 0.6$ and $\sigma_{sh} = 0.05$ for all sequences. Lexcoht did not have any parameters requiring tuning.

The developed GDS is expected to be used repeatedly. Regarding convergence time, it is therefore appropriate to consider the frequency with which one wants the best technique to be faster than the others. In this test, it was decided that the best technique should generate faster solutions at least twice as often as the second best technique. In other words, 25% of the time the convergence time of the second-best technique should be below the median of the first technique (obviously, 50% of the time the convergence time of the first technique is below its median, i.e., twice as often as the second one). If the computing times of the techniques are normally distributed with the same standard deviation, then the mean is confounded with the median. In that case, the second-best mean should be at least 0.68 SD away from the best mean [$N(-0.68;0,1) = 0.25$]. The desired effect size is therefore $d = (d\mu/\sigma) = 0.68$. In Motte et al. (2011) the distributions were positively skewed; the chosen effect size is therefore quite conservative. With 13 treatments to compare against each other using the Tukey test, and with $d = 0.68$, the minimum number of runs for each treatment is 48 (Nicewander & Price, 1997). To control for nonconvergence (estimated originally at 10%), the chosen number of runs was set at 60. Finally, a repeated-measure design was used, allowing for studying whether the original populations had an effect on diversity for the different techniques.

The performed simulation presented low convergence rates for the BM techniques (40%–66%). This was unexpected, because the BM technique had always converged in the previous study (Motte et al., 2011) and had high convergence rates in an unpublished prestudy. As the previous simulation was based on a repeated-measures design, it was not possible to exploit it for convergence rate and convergence time, because of the large number of missing data. Therefore, the original simulation based on repeated-measures design was used only for investigating diversity, and a new simulation was performed under the same conditions, with independent samples for the convergence time and the convergence rate. The number of runs in each treatment was set at 150 in order to ensure a sufficient power.

The convergence times of the treatments were obtained using the CPU time of one core of an Intel Xeon E5620 2.40 GHz processor. The total simulation time amounted to 22 days, 13 h (because three CPU cores were used simultaneously, the simulations took 254 h).

3.5. Results

3.5.1. Diversity

The diversity within a population (or "intrapopulation diversity") is calculated by the sum of all individuals' diversities.

The intrapopulation diversity among all treatments was 1.24×10^{-2} (SD $= 4.48 \times 10^{-2}$). The intrapopulation diversity for each method and technique is reported in Table 2.

The alternatives offered by the methods did not present an appreciable variety until the diversity reached a value of 8×10^{-2} (e.g., Fig. 3). The alternatives with a diversity between 8×10^{-2} and 9×10^{-2} are in a gray area (see Fig. 4), while the alternatives above 9×10^{-2} are clearly dissimilar (Fig. 5). Unfortunately, only four pairs of different individuals for all the methods and sequences had a diversity value between 8×10^{-2} and 9×10^{-2} (two in population 27 of the BM *laf* sequence, two in population 30 of the same sequence, two in population 27 of the BM *lfa* sequence, and two in population 27 of the UWS method). The different variants with a diversity above 9×10^{-2} were also few. For Lexcoht, one population in each of the *alf* and *lfa* sequences presented 2 alternatives that could be judged as diverse. For the BM method, one population of the *fal* sequence presented 2 alternatives, 3 populations of the *fla* sequence presented 2 (2 and 14 alternatives respectively), 2 populations of the *laf* sequence presented 2 and 6 alternatives, and 2 populations of the *lfa* sequence presented 2 and 7 alternatives. The UWS method did not have any variant above 9×10^{-2}. These outcomes are summarized Table 3. The probability of getting dissimilar individuals in one population is therefore not only very low; the number of dissimilar alternatives per population is also generally low: most of the time, the user is not expected to obtain more than 2 alternatives. Moreover, most of the dissimilar alternatives originate from the same original populations (populations 27, 30, 43; see Table 3). The diversity seems to depend more on the good characteristic of the original population than on the method itself. The BM method did get most of the dissimilar groups, but not as much as was expected. The sharing scheme of the BM method probably does not create diversity among individuals but seems to maintain it if it is present in the original population.

The intrapopulation diversity did not provide satisfying results. However, the diversity between populations (or "interpopulation diversity") was much larger: 1.63×10^{-1} (SD $= 6.85 \times 10^{-2}$). The minimal diversity value was $0.802 \times$

Table 2. *Intrapopulation diversity measures (standard deviations)*

Treatment	Diversity (SD)
L-*afl*	3.69×10^{-2} (1.43×10^{-2})
L-*alf*	8.20×10^{-3} (4.54×10^{-3})
L-*fal*	6.42×10^{-3} (4.45×10^{-3})
L-*fla*	8.82×10^{-3} (6.93×10^{-3})
L-*laf*	1.04×10^{-3} (1.32×10^{-3})
L-*lfa*	2.04×10^{-2} (6.19×10^{-3})
BM-*afl*	1.54×10^{-3} (2.88×10^{-3})
BM-*alf*	5.99×10^{-3} (4.56×10^{-3})
BM-*fal*	1.77×10^{-2} (6.00×10^{-3})
BM-*fla*	6.35×10^{-2} (1.38×10^{-2})
BM-*laf*	4.36×10^{-2} (1.73×10^{-2})
BM-*lfa*	4.72×10^{-2} (1.45×10^{-2})
UWS	1.55×10^{-2} (9.20×10^{-3})

Note: L, Lexicoht; *laf*, *lfa*, *alf*, *afl*, *fla*, *fal*, the order in which the constraints are evaluated; BM, behavioral memory; UWS, unweighted sum scheme.

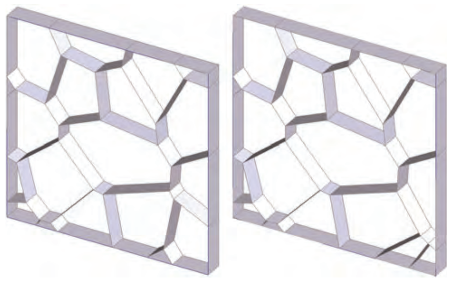


Fig. 3. (Color online) Two table support variants of population 23 for the Lexcoht *alf* sequence, with diversity 2.96×10^{-2}.

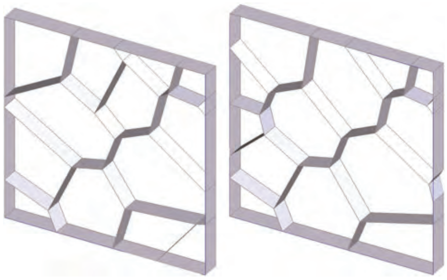


Fig. 4. (Color online) Two table support variants of population 27 for the unweighted sum scheme technique, with diversity 8.76×10^{-2}.

10^{-2}. For Lexcoht, there were only two populations of the *alf* sequence that contained individuals with a diversity below 9×10^{-2} and two populations in the *fla* sequence that contained individuals with a diversity below 9×10^{-2}. All the remaining individuals were quite dissimilar. *Running several simulations with different original populations therefore ensures diversity.*

Figure 6 illustrates the large difference between intra- and interpopulation diversities. It is very important that there is no need, at least in this particular example, to even measure diversity, because virtually all interpopulation individuals are dissimilar. The computing time becomes a function of the number of alternatives one wants to present to the user; however, the time taken to ensure interpopulation diversity by any future method is likely to consume additional time. Moreover, these additional simulations can run completely in parallel and, with the generalization of multicore servers, the running time would be virtually the same and depend only on the availability of computing resources.

3.5.2. Convergence time

The smallest convergence rate observed in the second simulation was 28%, which amounted to 42 successful runs. This is less than the required minimum number of runs for each treatment (48; see Section 3.4), which implies a loss of power but also a decrease in type I error, which means that the multiple-comparison test is rather conservative. It was therefore decided to go on with the obtained data. The exploratory data analysis revealed that the distributions of the convergence time for each combination were markedly positively skewed, as is illustrated in Figure 7. The standard deviations were found proportional to the means; thus a logarithmic transformation was applied to the data (Howell, 2007, pp. 319−321). The log-transformed populations were mostly normally distributed; the Jarque–Bera test for normality (Jarque & Bera, 1987) failed to show a significant deviation from a normal distribution for most of the combinations (five treatments had $p_{\mathrm{JB}} < 0.01$). With the largest variance ratio being 1:4, the heteroscedasticity was within the limit on heterogeneity of variance (i.e., less than or equal to a factor of 4) for which the analysis of variance is still robust (Wilcox, 1987; Howell, 2007, p. 317).

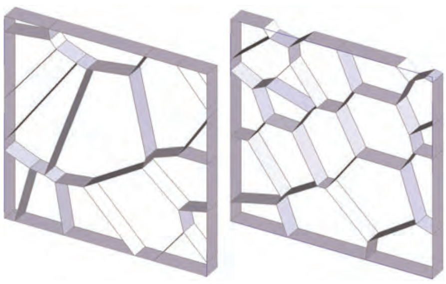


Fig. 5. (Color online) Two table support variants of population 37 of the *afl* sequence of Lexcoht, with diversity 1.87×10^{-1}.

Table 3. *Groups of alternatives with a diversity value above 8×10^{-2}*

Treatment	0.08–0.09	>0.09
L-*afl*	—	37:2
L-*alf*	—	—
L-*fal*	—	—
L-*fla*	—	—
L-*laf*	—	—
L-*lfa*	—	44:2
BM-*afl*	—	—
BM-*alf*	—	—
BM-*fal*	—	8:2
BM-*fla*	—	17:2, 30:14, 55:2
BM-*laf*	27:2, 30:2	30:6, 43:2
BM-*lfa*	27:2	30:7, 43:2
UWS	27:2	—

Note: The first value is the population from which the groups originate, and the second value is the number of dissimilar groups. L, Lexicoht; *laf*, *lfa*, *alf*, *afl*, *fla*, *fal*, the order in which the constraints are evaluated; BM, behavioral memory; UWS, unweighted sum scheme.

A one-way analysis of variance revealed that there were significant differences among the means of the 13 treatments [F (12, 1286) = 17.98, $p < 0.001$]. A Tukey test at $\alpha = 0.05$ was subsequently performed upon the 13 treatments. Figure 8 presents the log-transformed means for each method and sequence.

The Lexcoht method with the *alf* sequence was significantly better than the UWS method. It was not significantly better than the best BM method result, with the *laf* sequence, which itself was not significantly better than the UWS method.

The Lexcoht method with the *alf* sequence was significantly better than the worst Lexcoht sequence *fla*. The BM method with the best sequence (*laf*) was also significantly better than the worst BM sequence (*fal*). This confirms that the choice of the right sequence is important.

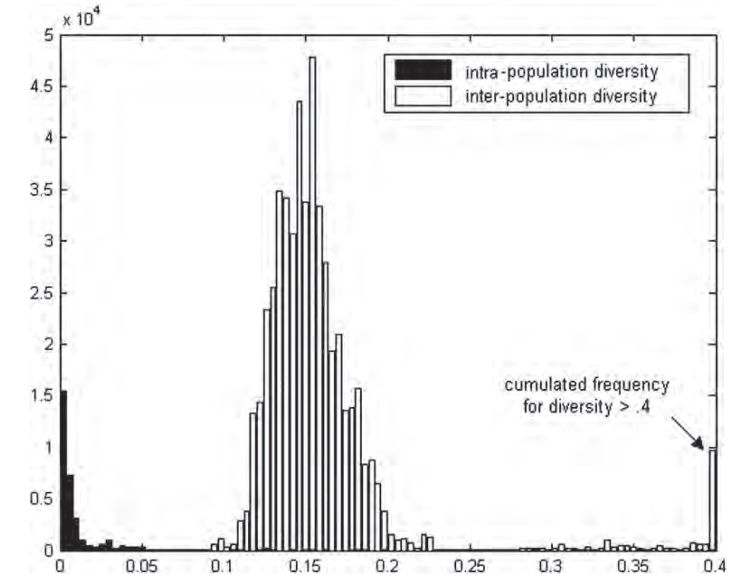


Fig. 6. A histogram of the intra- and interpopulation diversities. The interpopulation diversity is almost always superior to the intrapopulation diversity.

3.5.3. Convergence rate

The convergence rates of the different treatments are presented Table 4. A chi-square test for proportions produced $\chi^2\,(12) = 464.02$, which is significant at $p < 0.001$. A pairwise comparison following the Tukey–Kramer procedure for proportions (Hochberg & Tamhane, 1987, p. 275) was subsequently performed. The convergence rates of Lexcoht with sequences *afl* and *fal* were significantly larger than the other treatments. The complete results are presented Table 4 and Figure 9. Almost all Lexcoht treatments have a significantly higher convergence rate than the BM treatments. The BM treatments with a computing time similar to the best Lexcoht results as well as UWS are therefore performing significantly worse in terms of convergence rate.

3.6. Discussion

In this paper, a number of different techniques for handling technical constraints have been evaluated in terms of convergence time, convergence rate, and the diversity of the generated solutions using a real product design problem. The aim has been to investigate generative product design systems used in the context of mass customization, which are required to quickly and reliably generate diverse solutions without requiring modification or tuning during use. When such systems are designed to allow for the customization of product form, they must be able to handle production and engineering constraints that can be time-consuming to evaluate and difficult to fulfill. These issues are related to how the constraints are handled in the GDS, and because of this, two promising SCHTs and the often used weighted sum technique have been investigated.

Concerning diversity, the investigation revealed that the intrapopulation diversity was not high enough to be used for presenting several alternatives to the user. In contrast, the interpopulation diversity was always high. Diversity is thus gained at the cost of more runs, but in that case there is no need to check for diversity (as all interpopulation solutions are sufficiently different). This result is also important because, if generalized, it would imply that it is not even necessary to define a diversity measure, *whatever the type of complex form*. It could also be shown that the specific mating scheme that is built in in the BM method did not ensure enough intrapopulation diversity.

The treatments that were most frequently the fastest were, for Lexcoht, the *alf* sequence and, for BM, the *laf* sequence. Lexcoht with the best sequence outperformed UWS by a factor of two. Although this confirms that the SCHTs are promising for the kind of problem presented here, it does not completely rule out the UWS, which performed well for the investigated design problem and requires no tuning or

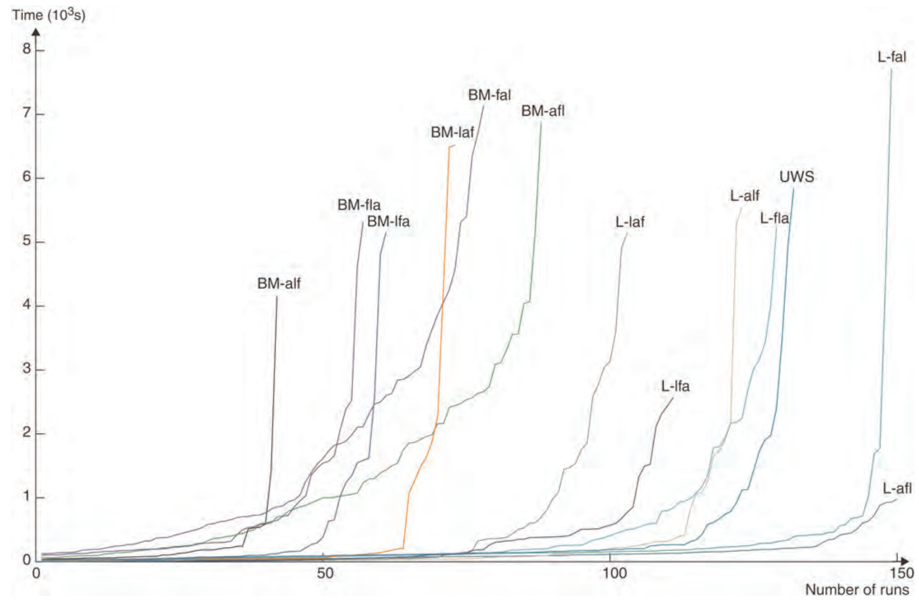


Fig. 7. (Color online) A representation of the sorted convergence times of the 13 treatments.

sequence selection. In the case of SCHTs the different sequences need to be tested first, but the gain is substantial if the GDS is used frequently. It is important to recall that the convergence time distributions are highly positively skewed, so a good CHT not only allows for a quicker convergence on average but also avoids very lengthy runs. The parameters for the BM techniques that were set according to the recommendations from Schoenauer and Xanthakis (1993) yielded good results.

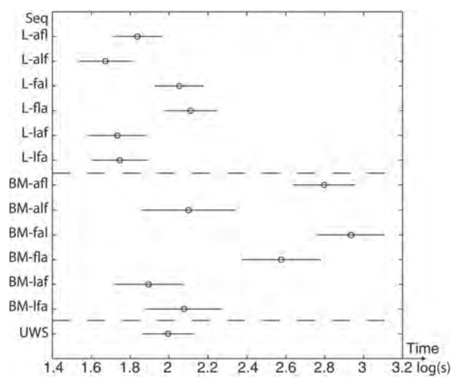


Fig. 8. A representation of the log-transformed means and their comparison intervals (95%) for the constraint-handling techniques.

Table 4. *Number of successful runs (out of 150), rate of convergence, and 95% Clopper–Pearson CI*

Treatment	*N*	Converg. Rate	95% CI Lower Bound	95% CI Upper Bound
L-*afl*	150	1.00	0.98	1.00
L-*alf*	123	0.82	0.75	0.88
L-*fal*	149	0.99	0.96	1.00
L-*fla*	129	0.86	0.79	0.91
L-*laf*	103	0.69	0.61	0.76
L-*lfa*	111	0.74	0.66	0.81
BM-*afl*	88	0.59	0.50	0.67
BM-*alf*	42	0.28	0.21	0.36
BM-*fal*	78	0.52	0.44	0.60
BM-*fla*	57	0.38	0.30	0.46
BM-*laf*	73	0.49	0.40	0.57
BM-*lfa*	61	0.41	0.33	0.49
UWS	132	0.88	0.82	0.93

Note: CI, confidence interval; L, Lexicoht; *laf*, *lfa*, *alf*, *afl*, *fla*, *fal*, the order in which the constraints are evaluated; BM, behavioral memory; UWS, unweighted sum scheme.

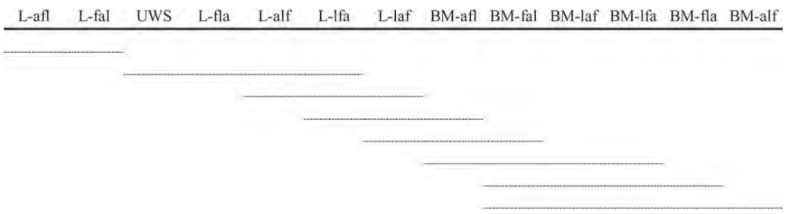


Fig. 9. A representation of the homogeneous populations (treatments ranked) for $p < 0.05$.

The treatments that had the best convergence rates were Lexcoht with the *afl* and *fal* sequence. The convergence rates were poor for the BM method in this setup but were excellent in Motte et al. (2011) at 100%. Note that convergence rate and time are not correlated (compare Figs. 8 and 9). Therefore, the choice of an adequate sequence must take into account convergence rate and time, as well as computing resources (the calculations can be made in parallel with multicore or cluster setups).

One should nevertheless remember that in order to use SCHTs, a good constraint sequence has to be found. This is a time-consuming task that requires a careful experimental design. As mentioned earlier, the presented comparison took around 10 days. This comparison is interesting only if the GDS is to be used frequently; otherwise the weighted sum is the best default technique.

4. CONCLUSION

The perspective of enabling consumers to use potentially complex forms, coupled to functional, engineering, and production constraints, is appealing. Several obstacles to such an approach have been dealt with in this article. Although much research has been done in the area of GDSs, few take into account constraints that are time-consuming to evaluate and difficult to fulfill, such as structural stability and manufacturability, a necessity for many products based on mass production systems. The ones that achieve that are focused on finding the best solution in regard to the objectives, rather than user preferences, and are not targeted at consumers. In order to give the consumer meaningful choice among the generated solutions, they must all fulfill the constraints and should be generated quickly and reliably to avoid frustration. These issues are all related to how the constraints are handled, and our aim has been to investigate how CHTs in a GDS intended to be used in the context of mass customization of product form should handle difficult constraints. In terms of CHTs, virtually all GDS applications dealing with more than one constraint or objective are applying the weighted sum technique. We have therefore evaluated three promising CHTs, two SCHTs and the UWS. The results show that the Lexcoht SCHT outperformed the UWS in terms of both convergence time and rate and that diversity can be guaranteed by launching many design generations in parallel. Enabling the efficient handling of such types of constraints is a step toward showing that form mass customization is technically possible, and beyond that a step toward total mass customization. Algorithmic form generation, coupled to an interactive compelling online experience as well as purchase, logistics, and production back-end, allows for various entrepreneurial opportunities for companies and consumers alike, as well as for the designers.

The scope of application of new digital means of interaction, designing, and fabrication is fully scalable and in that sense constitutes a unique enabler that, if consistently implemented, could potentially cut across a very large number of industries, ranging from small manufacturers to large producers of consumer products. The research presented in this paper addresses primarily the use of GDSs in the mass customization context, but some aspects of it, especially the diversity issue, should also be useful in a GDS intended for professional industrial designers.

ACKNOWLEDGMENTS

This work was supported by the Swedish Governmental Agency for Innovation Systems, VINNOVA, through the Production Strategies and Models for Product Development program (Project Renaissance 1.5, 2009-04057).

REFERENCES

Agarwal, M., & Cagan, J. (1998). A blend of different tastes: the language of coffeemakers. *Environment and Planning B 25(2)*, 205–227.

Agarwal, M., Cagan, J., & Constantine, C.G. (1999). Influencing generative design through continuous evaluation: associating costs with the coffeemaker shape grammar. *Artificial Intelligence for Engineering Design, Analysis and Manufacturing 13(4)*, 253–275.

Ang, M.C., Chau, H.H., McKay, A., & de Pennington, A. (2006). Combining evolutionary algorithms and shape grammars to generate branded product design. *Proc. 2nd Design Computing and Cognition Conf., DCC '06*, pp. 521–539. Dordrecht: Springer.

Austrell, P.E., Dahlblom, O., Lindemann, J., Olsson, A., Olsson, K.-G., Persson, K., Petersson, H., Ristinmaa, M., Sandberg, G., & Wernberg, P.-A. (2004). *CALFEM: A Finite Element Toolbox, Version 3.4*. Lund: Lund University, Structural Mechanics LTH.

Beukers, A., & van Hinte, E. (2005). *Lightness: The Inevitable Renaissance of Minimum Energy Structures*, 4th rev. ed. Rotterdam: 010 Publishers.

Bouché, N. (2009). Keynote V: how could we create new emotional experiences with sensorial stimuli? *Proc. 4th Int. Conf. Designing Pleasurable Products* *and Interfaces, DPPI '09*, p. 21. Compiègne, France: Université de Technologie de Compiègne.

Bouyssou, D., Marchant, T., Pirlot, M., Tsoukiàs, A., & Vincke, P. (2006). *Evaluation and Decision Models With Multiple Criteria: Stepping Stones for the Analyst*. New York: Springer.

Cagan, J. (2001). Engineering shape grammars: where have we been and where are we going? In *Formal Engineering Design Synthesis* (Antonsson, E.K., & Cagan, J., Eds.), chap. 3, pp. 65–92. Cambridge, MA: Cambridge University Press.

Chau, H.H., Chen, X., McKay, A., & de Pennington, A. (2004). Evaluation of a 3D shape grammar implementation. *Proc.1st Design Computing and Cognition Conf., DCC '04*, pp. 357–376. Dordrecht: Kluwer.

Cluzel, F., Yannou, B., & Dihlmann, M. (2012). Using evolutionary design to interactively sketch car silhouettes and stimulate designer's creativity. *Engineering Applications of Artificial Intelligence 25(7)*, 1413–1424.

Coello Coello, C.A. (2002). 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(11–12)*, 1245–1287.

Deb, K., & Goldberg, D.E. (1989). An investigation of niche and species formation in genetic function optimization. *Proc. 3rd Int. Conf. Genetic Algorithms, ICGA '89*, pp. 42–50. Los Altos, CA: Morgan Kaufman.

de Garis, H. (1990). Genetic programming: building artificial nervous systems with genetically programmed neural network modules. *Proc. 7th Int. Conf. Machine Learning*, pp. 132–139. Los Altos, CA: Morgan Kaufmann.

El-Sayed, A., & Chassapis, C. (2005). A decision-making framework model for design and manufacturing of mechanical transmission system development. *Engineering With Computers 21(2)*, 164–176.

EZCT Architecture & Design Research, Hamda, H., & Schoenauer, M. (2004). *Studies on Optimization: Computational Chair Design Using Genetic Algorithms*. Paris: EZCT Architecture & Design Research.

Frazer, J.H. (1996). The dynamic evolution of designs. *Proc. 4D Dynamics Conf. Design and Research Methodologies for Dynamic Form*, pp. 49–53. Leicester: De Montfort University, School of Design & Manufacture.

Friebe, H., & Ramge, T. (2008). *Marke Eigenbau: Der Aufstand der Massen gegen die Massenproduktion* [The DIY Brand: The Rebellion of the Masses Against Mass Production]. Frankfurt: Campus Verlag.

Goldberg, D.E., & Richardson, J. (1987). Genetic algorithms with sharing for multimodal function optimization. *Proc. 2nd Int. Conf. Genetic Algorithms, ICGA '87*, pp. 41–49. Hillsdale, NJ: Erlbaum.

Haug, E.J., & Arora, J.S. (1979). *Applied Optimal Design—Mechanical and Structural Systems*. New York: Wiley.

Hochberg, Y., & Tamhane, A.C. (1987). *Multiple Comparison Procedures*. New York: Wiley.

Holland, J.H. (1975). *Adaptation in Natural and Artificial Systems: An Introductory Analysis with Applications to Biology, Control and Artificial Intelligence*. Ann Arbor, MI: University of Michigan Press.

Howell, D.C. (2007). *Statistical Methods for Psychology*, 6th ed. Belmont, CA: Thomson Wadsworth.

Jarque, C.M., & Bera, A.K. (1987). A test for normality of observations and regression residuals. *International Statistical Review 55(2)*, 163–172.

Johansson, J. (2011). How to build flexible design automation systems for manufacturability analysis of the draw bending of aluminum profiles. *Journal of Manufacturing Science and Engineering 133(6)*, 1–11.

Johnson, L.M. (2012). *B-shelves: a web based mass customized product*. Master's Thesis. Seattle, WA: University of Washington, Department of Architecture.

Kaplan, A.M., & Haenlein, M. (2006). Toward a parsimonious definition of traditional and electronic mass customization. *Journal of Product Innovation Management 23(2)*, 168–182.

Knight, T.W. (1980). The generation of Hepplewhite-style chair-back designs. *Environment and Planning B 7(2)*, 227–238.

Kraft Foods. (2006). *Innovate With Kraft*. Accessed at http://brands.kraftfoods.com/innovatewithkraft/default.aspx on January 17, 2010.

Kram, R., & Weisshaar, C. (2003). Breeding tables. Accessed at http://www.kramweisshaar.com/projects/breeding-tables.html on February 22, 2012.

Lakhani, K.R., & Kanji, Z. (2009). *Threadless: The Business of Community*, Harvard Business School Multimedia/Video Case 608-707. Cambridge, MA: Harvard Business School.

Lee, H.C., & Tang, M.X. (2009). Evolving product form designs using parametric shape grammars integrated with genetic programming. *Artificial Intelligence for Engineering Design, Analysis and Manufacturing 23(2)*, 131–158.

Lin, B.-T., Chang, M.-R., Huang, H.-L., & Liu, C.-Y. (2009). Computer-aided structural design of drawing dies for stamping processes based on functional features. *International Journal of Advanced Manufacturing Technology 42(11–12)*, 1140–1152.

McCormack, J.P., Cagan, J., & Vogel, C.M. (2004). Speaking the Buick language: capturing, understanding, and exploring brand identity with shape grammars. *Design Studies 25(1)*, 1–29.

Mezura-Montes, E. (2004). *Alternative techniques to handle constraints in evolutionary optimization*. PhD Thesis. CINVESTAV-IPN, Electrical Engineering Department, Computer Science Section.

Mezura-Montes, E., & Coello Coello, C.A. (2006). *A Survey of Constraint-Handling Techniques Based on Evolutionary Multiobjective Optimization*, Technical Report EVOCINV-04-2006. CINVESTAV-IPN, Department of Computation, Evolutionary Computation Group.

Michalewicz, Z., Dasgupta, D., Le Riche, R.G., & Schoenauer, M. (1996). Evolutionary algorithms for constrained engineering problems. *Computers & Industrial Engineering 30(4)*, 851–871.

Michalewicz, Z., & Janikow, C.Z. (1991). Handling constraints in genetic algorithms. *Proc. 4th Int. Conf. Genetic Algorithms, ICGA '91*, pp. 151–157. San Mateo, CA: Morgan Kaufmann.

Michalewicz, Z., & Schoenauer, M. (1996). Evolutionary algorithms for constrained parameter optimization problems. *Evolutionary Computation 4(1)*, 1–33.

Moser, K., Müller, M., & Piller, F. (2006). Transforming mass customisation from a marketing instrument to a sustainable business model at Adidas. *International Journal of Mass Customisation 1(4)*, 463–479.

Motte, D., Nordin, A., & Bjärnemo, R. (2011). Study of the sequential constraint-handling technique for evolutionary optimization with application to structural problems. *Proc. 37th Design Automation Conf., DETC/DAC '11*, pp. 521–531. Washington, DC: ASME.

Nicewander, W.A., & Price, J.M. (1997). A consonance criterion for choosing sample size. *American Statistician 51(4)*, 311–317.

Nordin, A., Hopf, A., Motte, D., Bjärnemo, R., & Eckhardt, C.-C. (2011). Using genetic algorithms and Voronoi diagrams in product design. *Journal of Computing and Information Science in Engineering 11(1)*, 1–7.

Orsborn, S., Cagan, J., & Boatwright, P. (2008). Automating the creation of shape grammar rules. *Proc. 3rd Design Computing and Cognition Conf., DCC '08*, pp. 3–22. Dordrecht: Springer Science + Business Media.

Orsborn, S., Cagan, J., Pawlicki, R., & Smith, R.C. (2006). Creating cross-over vehicles: defining and combining vehicle classes using shape grammars. *Artificial Intelligence for Engineering Design, Analysis and Manufacturing 20(3)*, 217–246.

Paredis, J. (1994). Co-evolutionary constraint satisfaction. *Proc. 3rd Parallel Problem Solving from Nature Conf., PPSN III*, LNCS, Vol. 866, pp. 46–55. Berlin: Springer.

Pearce, P. (1978). *Structure in Nature Is a Strategy for Design*. Cambridge, MA: MIT Press.

Petersson, H., Motte, D., Eriksson, M., & Bjärnemo, R. (2012). A computer-based design system for lightweight grippers in the automotive industry. *Proc. Int. Mechanical Engineering Congr. Exposition, IMECE '12*. Houston, TX: ASME.

Piasecki, M., & Hanna, S. (2010). A redefinition of the paradox of choice. *Proc. 4th Design Computing and Cognition Conf., DCC '10*, pp. 347–366. Dordrecht: Springer.

Pugliese, M.J., & Cagan, J. (2002). Capturing a rebel: modeling the Harley-Davidson brand through a motorcycle shape grammar. *Research in Engineering Design 13(3)*, 139–156.

Sandberg, M., & Larsson, T. (2006). Automating redesign of sheet-metal parts in automotive industry using KBE and CBR. *Proc. 32nd Design Automation Conf., DETC/DAC '06*, pp. 349–357. Philadelphia, PA: ASME.

Schoenauer, M., & Michalewicz, Z. (1996). Evolutionary computation at the edge of feasibility. *Proc. 4th Parallel Problem Solving From Nature Conf., PPSN IV*, LNCS, Vol. 1141, pp. 245–254. Berlin: Springer.

Schoenauer, M., & Xanthakis, S. (1993). Constrained GA optimization. *Proc. 5th Int. Conf. Genetic Algorithms, ICGA '93*, pp. 573–580. San Mateo, CA: Morgan Kaufmann.

Shea, K., & Cagan, J. (1999). Languages and semantics of grammatical discrete structures. *Artificial Intelligence for Engineering Design, Analysis and Manufacturing 13(4)*, 241–251.

Toffler, A. (1971). *Future Shock*, 2nd ed. New York: Bantam Books.

Trubridge, D. (2010). “Coral [lamp].” Accessed at http://www.davidtrubridge.com/coral/ on January 29, 2011.
Van Le, T. (1996). A fuzzy evolutionary approach to constrained optimisation problems. *Proc. 3rd IEEE Int. Conf. Evolutionary Computation, ICEC ’96*, pp. 274–278. Piscataway, NJ: IEEE.
Wenli, Z. (2008). Adaptive interactive evolutionary computation for active intent–oriented design. *Proc. 9th Int. Conf. Computer-Aided Industrial Design and Conceptual Design, CAIDCD ’08*, pp. 274–279. Piscataway, NJ: IEEE.
Wilcox, R.R. (1987). *New Statistical Procedures for the Social Sciences*. Hillsdale, NJ: Erlbaum.
Yeniay, Ö. (2005). Penalty function methods for constrained optimization with genetic algorithms. *Mathematical and Computational Applications 10(1)*, 45–56.

Axel Nordin is a PhD student at the Division of Machine Design, Lund University. He received his MS in industrial design engineering from Lund University and has participated in two government-funded research projects during the last 2 years. His work has been concerned mainly with studying aspects of integrating complex morphologies into the design of bespoke products, such as computational, manufacturing, structural, and usability challenges.

Damien Motte currently holds a postdoctoral position at the Division of Machine Design, Lund University. He received a PhD from the same division, a research master’s from the Industrial Engineering Laboratory at the École Centrale Paris, and an MS in industrial engineering from the École des Mines d’Albi, France. Dr. Motte is currently working on alternative engineering design and product development methodologies.

Andreas Hopf is a Design Consultant and Senior Lecturer in the School of Industrial Design at Lund University. He received a BA in industrial design from Art Center College of Design (Europe), Switzerland. At Lund University he runs seminars in industrial design and computer-aided design in two-dimensional/three-dimensional and rapid prototyping, supervises diploma projects, and participates in the development of new bachelor’s and master’s programs.

Robert Bjärnemo is a Professor of machine design at Lund University. He obtained his MS and PhD from the same university. Dr. Bjärnemo’s research interests are in engineering design methodology and product development methodology, especially integrated product development, as well as predictive design analysis.

Claus-Christian Eckhardt has been a Professor of industrial design at Lund University since 2001. He worked as an Interior Designer for Silvestrin Design and was in charge of designing consumer electronics and communication products at Blaupunkt, where he was also responsible for the design of the Bosch Telecom product series and Bosch mobile phones. Mr. Eckhardt later became Chief Designer and Head of Global Product Design with Bosch and then Head of Design at Tenovis and Avaya. He has also worked as a Design Consultant since 2000. Claus-Christian is the recipient of several national and international awards and honors. His research areas are in design management and design implementation.

Paper VII

Restart strategies for constraint handling in generative design systems

VII

Nordin, A.

40th Design Automation Conference - DETC/DAC'14,
Buffalo, NY, August 17-20, 2014

Originally published by ASME

Proceedings of the ASME 2014 International Design Engineering Technical Conferences & Computers and Information in Engineering Conference
IDETC/CIE 2014
August 17-20, 2014, Buffalo, New York, USA

DETC2014-34858

RESTART STRATEGIES FOR CONSTRAINT-HANDLING IN GENERATIVE DESIGN SYSTEMS

Axel Nordin
Division of Machine Design
Department of Design Sciences LTH
Lund University
Box 118, 221 00 Lund
Sweden
Email: axel.nordin@mkon.lth.se

ABSTRACT

Product alternatives suggested by a generative design system often need to be evaluated on qualitative criteria. This evaluation necessitates that several feasible solutions which fulfill all technical constraints can be proposed to the user of the system. Also, as concept development is an iterative process, it is important that these solutions are generated quickly; i.e., the system must have a low convergence time. A problem, however, is that stochastic constraint-handling techniques can have highly unpredictable convergence times, spanning several orders of magnitude, and might sometimes not converge at all. A possible solution to avoid the lengthy runs is to restart the search after a certain time, with the hope that a new starting point will lead to a lower overall convergence time, but selecting an optimal restart-time is not trivial. In this paper, two strategies are investigated for such selection, and their performance is evaluated on two constraint-handling techniques for a product design problem. The results show that both restart strategies can greatly reduce the overall convergence time. Moreover, it is shown that one of the restart strategies can be applied to a wide range of constraint-handling techniques and problems, without requiring any fine-tuning of problem-specific parameters.

INTRODUCTION

Within product development projects, many activities may require several iterations before a solution that fulfills engineering constraints and design specifications can be found. During the concept development activity, it is important to be able to quickly evaluate the technical aspects of a product proposal and generate new designs based on this evaluation.

A possible aid in this process is a generative design system (GDS) that generates potential product designs, while leaving the designer in control of the final design selection. GDSs have been developed, for example, to help the designer preserve the "form identity" of a brand [1-3]. A GDS intended for product design is basically structured around a graphical user interface and is often coupled to an interactive optimization system or a constraint satisfaction system that handles user preferences and technical constraints such as production and functional constraints. Through the interface, the designer can evaluate, select and influence the generation of designs.

A hurdle commonly encountered in association with GDSs is that the decision to choose one design over another is often not based on pure performance metrics, but rather on criteria that are subjective and difficult to quantify and thus left to the designer to evaluate. In order to give the designer a meaningful choice, the designs generated by a GDS need to fulfill all technical constraints, which may be time-consuming to evaluate and hard to satisfy. An efficient method for handling constraints is therefore an integral part of a GDS.

In preceding studies [5;16], several constraint-handling techniques (CHTs) based on genetic algorithms were evaluated in terms of the time needed to converge to a solution to engineering design problems. The results showed that the convergence times varied between several orders of magnitude, and were surprisingly unpredictable, even for stochastic methods such as genetic algorithms. The means by which the discovered variability of the convergence time can be exploited is the subject of this paper. In order to avoid spending large computational resources on searches that might have very long, or even infinite convergence times, a threshold, or a cutoff value, for when to restart the search can be set, with the hope that a new, randomly selected, starting point will lead to a lower overall convergence time.

However, determining an ideal cutoff value is not trivial; selecting a low cutoff value decreases the probability of finding a solution within a single search, which increases the number of iterations necessary for convergence. Moreover, as the search is conducted on an engineering problem, where the probability to converge to a solution within a certain convergence time is unknown, it is not possible to find the optimal cutoff value analytically. Therefore, an algorithm for determining when to restart the search needs to be either independent of the problem, or be able to adapt the cutoff value as the search progresses.

The benefit of using an adaptive strategy is that it can utilize information about successful and unsuccessful searches to gradually improve the approximation of the optimal cutoff value, whereas a problem-independent strategy does not improve with time.

In this paper, one adaptive strategy is presented for determining the cutoff value, and is compared it in terms of convergence time to a problem-independent strategy suggested by Luby *et al*. [6]. The two restart strategies are applied to two baseline CHTs that do not employ restarting. As the restart strategies do not rely on any prior knowledge of the problem, they can be applied to a broad range of constraint satisfaction problems with minimal adjustment.

The results show that restarting the search leads to a significant reduction in the convergence time for both restart strategies for the given application, with the adaptive strategy performing better than the distribution-independent strategy.

RESTART STRATEGIES

The emergence of restart strategies is mainly due to the discovery of problems, or rather runtime distributions (RTDs), that are highly unpredictable and exhibit a heavy tail of very long or infinite runtimes (see [7-9]). While heavy-tailed RTDs are generally detrimental to the efficiency of a CHT, they can also be exploited to provide substantial speedups by the use of restart strategies. In order for the restart strategy to be efficient, it is necessary to determine the optimal cutoff value. In Gomes and Sabharwal [10], a summary is given of the general concepts behind restart strategies and the cutoff value's effect on the runtime. Further investigations of how the cutoff value affects the runtime in both serial and parallel cases is given by Shylo *et al.* [11]. A more formal foundation is given by Luby *et al.* [6], who investigates two approaches based on either using a single uniform cutoff value, i.e. the same cutoff value is used for all restarts, or a universal sequence of cutoff values. Luby shows that the uniform strategy is optimal for Las Vegas algorithms, but requires the RTD to be known in order to find the correct cutoff value. Determining the RTD analytically is, however, most often not possible. Rather, a number of sample runtimes on which to base an approximation are required. Using sample runs to train the restart strategy has been investigated in, for instance, [12]. It is also possible to use an on-line learning algorithm, which does not rely on a training set, to progressively improve on the estimation of the uniform cutoff value. In [13], Gagliolo and Schmidhuber use the converged and cutoff runtimes from a universal strategy to train a uniform strategy by a bandit approach with promising results.

Depending on the application, the sampling of runtimes may not be feasible. The runtime of a single converged solution might be very long, or the RTD might be so unpredictable that vast amounts of samples are needed to get a good approximation. To avoid this problem, Luby *et al.* [6] instead suggests a universal strategy which requires no information about the RTD. The universal strategy is based on an exponentially increasing but repeating sequence of cutoff values (1 1 2 1 1 2 4 1 1 2 1 1 2 4 8 …), which he shows to result in runtimes that are less than or equal to $192 l_p(\log(l_p) + 5)$, where l_p is the expected optimal running time.

A variation of the universal sequence is to instead scale the cutoff value by a factor after each restart or to use a linearly increasing cutoff value. Huang [14] compared six restart strategies to Luby's universal strategy on a number of boolean satisfiability benchmarks and found that Luby's strategy outperformed the others.

IMPLEMENTATION OF THE RESTART STRATEGIES

In this paper, two strategies are compared based on the results reported in the previous section. The first strategy is based on Luby's universal strategy, and the second is an adaptive uniform strategy.

Universal strategy

Luby's sequence of cutoff values (t_1, t_2, t_3, …) can more formally expressed as

$$t_i = \begin{cases} 2^{k-1} & \text{if } i = 2^k - 1, \\ t_{i-2^{k-1}+1} & \text{if } 2^{k-1} \leq i < 2^k - 1, \end{cases}$$

where k is any positive integer fulfilling either of the two conditions. While the universal approach does not require any information about the problem, the overhead of restarting the search needs to be taken into account, and in practice the cutoff values in the sequence are multiplied by a factor (see [14;15]). In this paper, the scaling factors for the two CHTs are determined by measuring the convergence times of a number of trial runs while varying the scaling factor and selecting the scaling factor that gives the lowest convergence time. The cutoff values in this implementation are based on time rather than iterations, but the generations of the genetic algorithm or the number of individuals evaluated could also have been used.

Adaptive uniform strategy

The uniform strategy is based on using a single cutoff value for all restarts (t, t, t, …). For the uniform strategy to be efficient, the optimal cutoff value must be determined based on the actual or estimated RTD, from which the cumulative distribution function *F(T)* can be calculated for any given cutoff value *T*. As shown in [6] and [13], the expected value of the total runtime t_T for a certain cutoff value can then be expressed as

$$E(t_T) = \frac{T - \int_0^T F(\tau)d\tau}{F(T)}$$

By either analytical or numerical minimization of $E(t_T)$, an optimal value of T can be found for the given RTD.

In this study, the initial runtimes on which the estimation of the RTD is based are collected in a training phase by simply running the CHT until five runs have converged. During the training phase, the adaptive uniform strategy performs identically to the non-restart CHT. It would be possible to apply a scheme such as in [13] to collect the initial data; but, for the sake of comparison between the universal and uniform strategies, this was not implemented.

In the adaptive uniform strategy described in this paper, the data collected from the initial runs is used to fit a non-parametric piecewise linear approximation of *F(T)*, which is then updated with each new convergence time collected. A numeric evaluation of $E(t_T)$ for different values of T is then performed, and the value of T that minimizes $E(t_T)$ is used as the next cutoff value. The approximation of *F(T)* could also have been based on a more complex regression model such as Kriging or could have been assumed to fit some predetermined polynomial or rational function; however, the fitting time, robustness and simplicity of the piecewise linear model has been favored in this application. The time required for fitting *F(T)* is negligible in comparison to the runtime of one iteration of the CHT.

THE STUDY

Objective of the study

As discussed in the introduction, several solutions that fulfill all technical constraints are usually requested by the user of a GDS in order to have a wide selection. However, unlike many of the benchmarks and problems studied in conjunction with restart strategies previously, the solution-space of a product design problem is often too large to exhaust, and the design parameters are usually continuous, making it unfeasible to find all the solutions to a given design problem. Therefore, the main performance metric of a CHT for a GDS is how quickly it can find many, but not all, solutions to a design problem. To best evaluate this metric, the cumulative time needed to find unique solutions was measured for each restart strategy, rather than comparing single convergence times. By letting the GDS find a relatively high number of solutions, data can be collected on how the restart strategies perform both when generating few solutions and when generating many solutions.

To investigate how the two restart strategies perform on RTDs with different features, two baseline CHTs were used to find solutions to a design problem. The first baseline CHT is easy to implement and requires no fine-tuning, but it has highly unpredictable convergence times, i.e. its RTD is heavy-tailed. The second baseline CHT requires careful set-up, but it converges quickly and reliably, i.e. its RTD is relatively uniform.

The objective of this study is thus to investigate how the universal and adaptive uniform restart strategies perform when used in conjunction with two CHTs with different features on a typical product design generation problem.

Problem

Design problem

The majority of the works published concerning restart strategies has been focused on discrete constraint satisfaction problems and boolean satisfiability problems. This study instead investigates how these strategies can be applied to continuous variable problems with actual production and functional constraints. A suitable design problem, which has been shown to produce long-tailed RTDs is described in [16]. The design problem is based on a GDS for generating table structures (see Figure 1 and Figure 2) based on a complex tessellation that must satisfy three production and structural constraints. The user of the GDS inputs design parameters such as the height of the table and the contour of the top. The GDS finds a number of design candidates that fulfill all constraints and present them to the user, who can then decide to choose one design, request more design candidates or re-launch the design generation with new inputs. The manufacturing processes used are laser cutting and CNC sheet metal bending. The geometry of the bending machine limits the flange lengths of the cells to be manufactured to never be shorter than 30 mm, and the bending angles to never be less than 35°. The structural requirements limit the maximum vertical displacement of any part of the table to 2.5 mm. Initially, there are no design objectives; the goal of the GDS is to quickly find several feasible designs which can be presented to the user to select from. The design parameters ruling the geometry of the support structure consists of 140 continuous values, leading to a vast design space. The output from the GDS has been validated by producing a number of tables based on the solutions. The GDS and design problem is described in depth in [17].

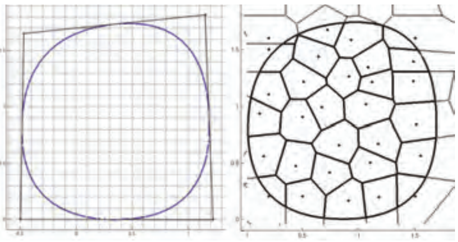


Figure 1. The interface through which the user of the GDS can define the table contour and review generated design candidates

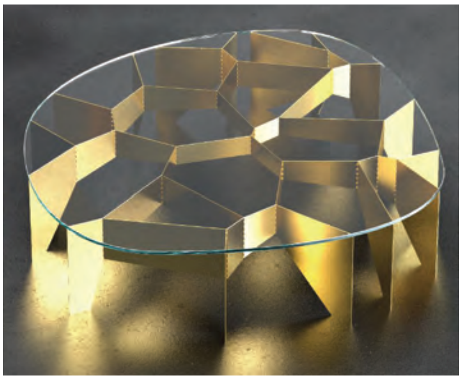


Figure 2. Example of a table structure generated by the GDS on which the restart strategies are applied

CHTs

The two CHTs used as baselines are based on the un-weighted sum (UWS) and the lexicographic constraint-handling technique (Lexcoht) from [16].

UWS is straightforward to implement and requires no tuning, yet performs equally well as weighted sums on this type of problem [5]. However, as shown in Figure 3, its RTD exhibits a heavy tail.

Lexcoht is based on handling the constraints in a lexicographic order, i.e., the constraints are handled in a defined sequence. As shown in [5] and [16], the order of the constraints heavily influences the runtime. With good choice of constraint sequence, Lexcoht outperforms UWS. The sequence used for Lexcoht in this paper was shown to have a high convergence rate and a rather flat RTD, as can be seen in Figure 3. Note that there is an order of magnitude difference in the span of the two RTDs.

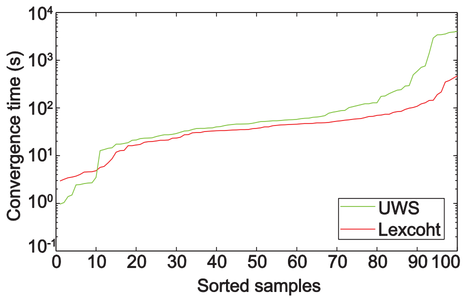


Figure 3. A sorted sample of 100 convergence times measured for UWS and Lexcoht

Experimental setup

Ten runs were executed for each combination of restart strategy and CHT. A total of 250 unique design candidates were requested in each run. The uniqueness was assured by comparing the geometry of the generated table structures, but no threshold for how similar two solutions could be was set. An evaluation of the diversity of the solutions is presented in the section "diversity". The input to the GDS was the same in every run. The measured runtimes were kept for the adaptive uniform strategy during the entire search for the 250 design candidates, but were reset between each of the ten runs.

The scaling factor used in the universal strategy was empirically determined based on the convergence times from 100 trial runs of the two CHTs while solving the design problem mentioned earlier. A larger sample size could potentially have yielded a better approximation of the optimal scaling factor, but the variation in convergence time is quite high and sample-sizes approaching the number of requested design candidates were deemed unfeasible. The optimal factor was determined to be .15 for UWS and 2.57 for Lexcoht, corresponding to approximately 2 and 30 calls to the evaluation function of the CHTs. As can be seen in Figure 4 and Figure 5, the scaling factor and the characteristics of the CHT's RTD greatly affects the performance of the universal strategy.

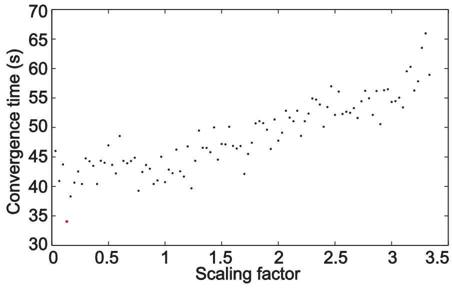


Figure 4. Sampled convergence times for the universal strategy when applied to UWS

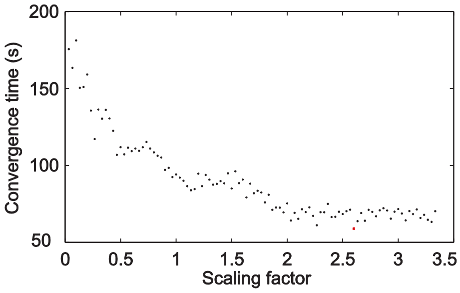


Figure 5. Sampled convergence times for the universal strategy when applied to Lexcoht

RESULTS AND DISCUSSION

This study evaluated the effectiveness of cutting off lengthy constraint satisfaction runs. Two strategies for determining when to cut off the current run and restart the search were studied. The first strategy is adaptive and gradually improves its approximation of the optimal restart-time (adaptive uniform), whereas the second strategy is static and relies on a universal heuristic for determining when to restart (universal). The two strategies were compared to two baseline CHTs, which do not employ restarting.

As the constraint-handling techniques presented in this paper are intended to be used in a GDS, an important performance metric is how quickly they can find numerous solutions to a design problem. To best evaluate this metric, the cumulative time needed to find 250 unique solutions was measured for each method, rather than comparing single convergence times.

Results for UWS

As can be seen in Table 1 and Figure 7, both restart strategies lead to significant reductions in total convergence time compared to the first baseline CHT. Figure 7 shows the maximum, minimum and mean cumulative convergence times for each method in logarithmic scale. The adaptive uniform strategy and the universal strategy achieve a mean reduction in convergence time of 94% and 91%, respectively. Table 1 shows that the variance of the total convergence times is quite high for the three methods, most likely due to the unpredictability of the first baseline CHT. However, even the longest total convergence time measured for the restart strategies is 85% lower than the shortest time for the baseline CHT. It should also be noted that the adaptive uniform strategy does not perform as well as the universal strategy during the first third of the search, as can be seen in Figure 6. This can be attributed to the learning process requiring a certain amount of data before a good approximation of the optimal restart-time can be made. After the initial learning period, the relative performance of the two restart strategies does not change much, and similar results are to be expected if more solutions were requested. A possibility is to use the two strategies in conjunction, using the restart-times suggested by the universal strategy during the first part of the search, while training the adaptive uniform strategy on the data collected until a stable approximation has been found. Gagliolo and Schmidhuber have investigated a similar approach in [13].

Table 1. Total runtimes for the three methods when applied to UWS

Method	Mean (s)	STD (s)
UWS	105229	14035
Universal	9787	532
Adaptive	5886	1481

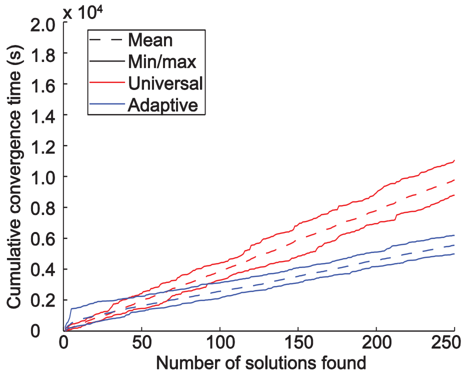


Figure 6. The two restarting strategies plotted separately for UWS

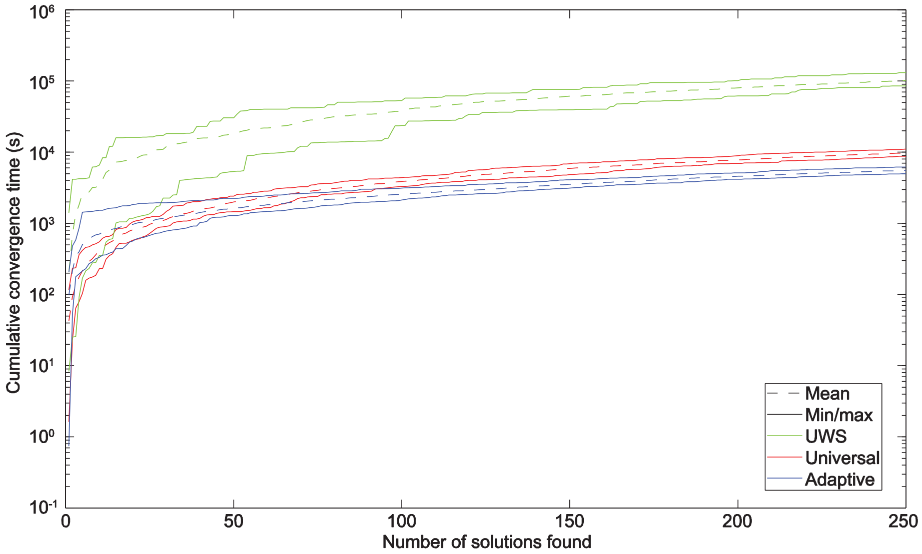


Figure 7. The cumulative convergence times for the three methods for the UWS CHT

Results for Lexcoht

The UWS baseline CHT is ideally suited for restart strategies as its convergence times span four orders of magnitude; however, not all CHTs behave in this way. The results for the second, more predictable and efficient baseline CHT, show how the restart strategies perform when the variance of the convergence time is low. As shown in Figure 9, even though the standard deviation of the runtimes from Lexcoht is an order of magnitude lower than for UWS, the adaptive uniform strategy is still able to find a restart-time that was high enough to avoid unnecessary restarts, resulting in a 24% reduction in the total convergence time compared to the baseline. The universal strategy is less suited for the problem and had a total convergence time that was 13% higher than the baseline. A comparison of the two restart strategies is shown in Figure 8.

Table 2. Total runtimes for the three methods for the when applied to Lexcoht

Method	Mean (s)	STD (s)
Lexcoht	14168	1024
Universal	16042	1029
Adaptive	10831	839

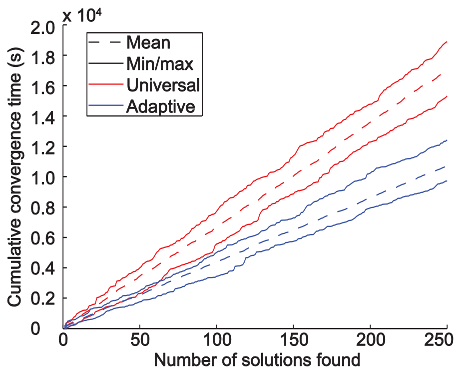


Figure 8. The two restarting strategies plotted separately for Lexcoht

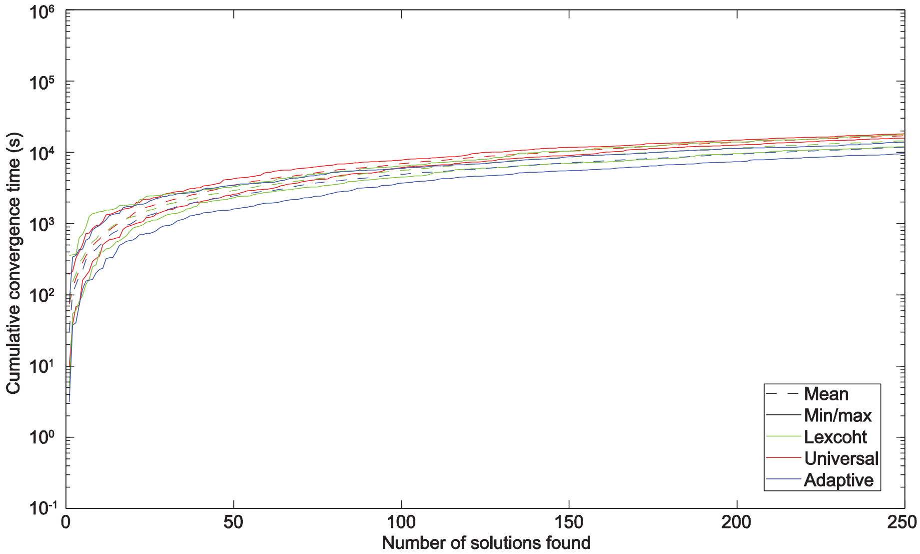


Figure 9. The cumulative convergence times for the three methods for Lexcoht

Diversity

Previous research [16] has focused on how the diversity of the solutions is affected by the choice of CHT. Due to the way the restart strategies favor more easily reachable solutions, the solutions could be quite similar, even though the results in [16] indicate a high diversity between solutions from separate runs (interpopulation diversity). In order to investigate how restarting affects the diversity of the solutions in this application, the diversity measure used in [16] was applied to the solutions. The diversity of an individual i to an individual j is calculated by comparing their genomes, in this case the coordinates of their Voronoi points. For each point a in individual i's genome, the point b in individual j's genome that has the smallest Euclidian distance d_a to point a is found, and the diversity measure is equal to the sum of d_a for all points in the genome.

As shown in Figure 10, the diversities are similar to the interpopulation diversities found in [16], with the lowest diversity being 0.078 and a mean diversity of 0.166. A sample of three pairs with low, medium and high diversity is shown in Figure 11.

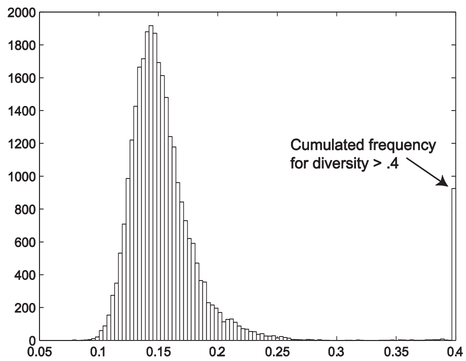


Figure 10. A histogram of the diversities of 250 solutions

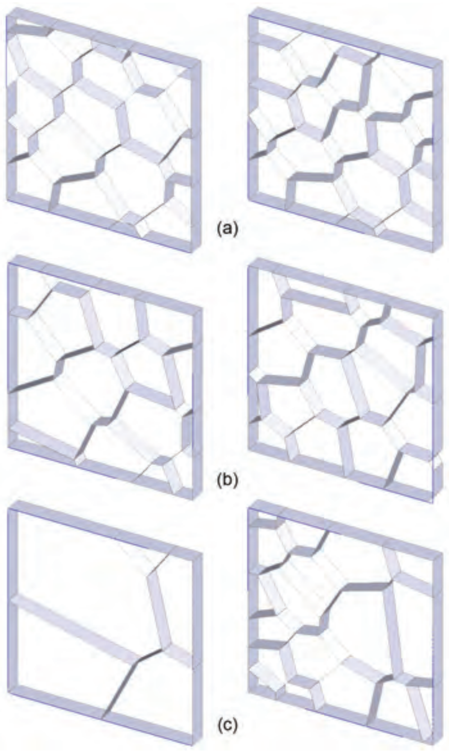


Figure 11. A sample of generated designs. The diversity measure for each pair is: *a*: 0.078, *b*: 0.151, *c*: 0.409

CONCLUSION

This study shows that the restart strategies significantly reduce the total convergence time compared to the heavy-tailed UWS CHT. Both restart strategies were shown to have strengths in different situations. The adaptive uniform strategy performed better overall, while the universal strategy had an advantage during the first third of the search. However, in order to get optimal results with the universal strategy, the scaling factor needs to be determined through some sort of heuristic beforehand, which might be time-consuming. In this study, the sample size used to approximate the optimal scaling factor was large in comparison to the number of requested solutions (100 samples for 250 solutions), although the results in Figure 4 and Figure 5 seem to indicate that a more sophisticated heuristic could have found a good scaling factor with much fewer samples. Additionally, Figure 4 and Figure 5 show that even a non-optimal scaling factor would have reduced the convergence time of UWS substantially.

The adaptive uniform strategy was able to reduce the total convergence time of the relatively flat-tailed Lexcoht, whereas the universal strategy did not show any improvements. The adaptability to different RTDs, and the lack of any parameters to be fine-tuned, should make the adaptive strategy applicable to any problem where stochastic search algorithms are used and efficient constraint-handling is important. The adaptive strategy is especially attractive if many solutions are sought, or if the GDS is to be used repeatedly with similar problems, as the information from previous runs can be used to achieve a better approximation of the RTD and thus the optimal cutoff value.

Comparing all the measured total convergence times reveals that both restart strategies, when used in combination with the first baseline CHT, perform better than the second baseline CHT both with and without restarting. This result is important as it shows that using restart strategies with the first baseline CHT, which is simple and generic, is more efficient than using the more complex CHT which requires careful set-up. A possible explanation can be found in the plot of the RTDs of UWS and Lexcoht in Figure 3, which shows that although the runtimes of UWS are generally high, a number of runtimes are actually much lower than those of Lexcoht, and can be exploited by the restarting strategies to lower the overall convergence time.

The evaluation of the diversity of the solutions showed that the restart strategies were not prone to finding the same solution repeatedly. The diversity was comparable to that of the baseline CHTs.

To further validate the generality and usefulness of restart strategies within GDSs, the RTDs of other design problems should be studied, and in particular how different user requirements affect the optimal cutoff value.

REFERENCES

[1] Pugliese, M. J. and Cagan, J., 2002, "Capturing a rebel: Modeling the Harley-Davidson brand through a motorcycle shape grammar", *Research in Engineering Design*, **13**(3), pp. 139-156.

[2] Chau, H. H., Chen, X., McKay, A. and de Pennington, A., 2004, "Evaluation of a 3D shape grammar implementation", *1st Design Computing and Cognition Conference - DCC'04*, Cambridge, MA, pp. 357-376.

[3] McCormack, J. P., Cagan, J. and Vogel, C. M., 2004, "Speaking the Buick language: Capturing, understanding, and exploring brand identity with shape grammars", *Design Studies*, **25**(1), pp. 1-29.

[4] Nordin, A., Motte, D. and Bjärnemo, R., 2013, "Strategies for consumer control of complex product forms in generative design systems", *39th Design Automation Conference - DETC/DAC'13*, Portland, OR.

[5] Motte, D., Nordin, A. and Bjärnemo, R., 2011, "Study of the sequential constraint-handling technique for evolutionary optimization with application to structural problems", *37th Design Automation Conference - DETC/DAC'11*, Washington, D.C., 5, pp. 521-531, Both authors contributed equally.

[6] Luby, M., Sinclair, A. and Zuckerman, D., 1993, "Optimal speedup of Las Vegas algorithms", *Information Processing Letters*, **47**(4), pp. 173-180.

[7] Gomes, C. P., Selman, B., Crato, N. and Kautz, H., 2000, "Heavy-Tailed Phenomena in Satisfiability and Constraint Satisfaction Problems", *Journal of Automated Reasoning*, **24**(1-2), pp. 67-100.

[8] Gomes, C. P., Selman, B. and Crato, N., 1997, "Heavy-tailed distributions in combinatorial search", in Smolka G. (Ed.), *Principles and Practice of Constraint Programming-CP97*, 1330 Edition, 1997, Springer Berlin Heidelberg, pp. 121-135.

[9] Hogg, T. and Williams, C. P., 1994, "The hardest constraint problems: A double phase transition", *Artificial Intelligence*, **69**(1-2), pp. 359-377.

[10] Gomes, C. P. and Sabharwal, A., 2009, "Exploiting runtime variation in complete solvers", in Biere A., Heule M., van Maaren H. and Walsh T. (Eds), *Handbook of Satisfiability*, 2009, IOS Press, Amsterdam, pp. 271-288.

[11] Shylo, O. V., Middelkoop, T. and Pardalos, P. M., 2011, "Restart strategies in optimization: parallel and serial cases", *Parallel Computing*, **37**(1), pp. 60-68.

[12] Kautz, H., Horvitz, E., Ruan, Y., Gomes, C. P. and Selman, B., 2002, "Dynamic restart policies", *The Eighteenth National Conference on Artificial Intelligence AAAI-02*, Edmonton, Canada, pp. 674-681.

[13] Gagliolo, M. and Schmidhuber, J., 2007, "Learning Restart Strategies", *International Joint Conference on Artificial Intelligence IJCAI-07*, Hyderabad, India, pp. 792-797.

[14] Huang, J., 2007, "The effect of restarts on the efficiency of clause learning", *International Joint Conference on Artificial Intelligence IJCAI-07*, Hyderabad, India, pp. 2318-2323.

[15] Audemard, G. and Simon, L., 2012, "Refining Restarts Strategies for SAT and UNSAT", in Milano M. (Ed.), *Principles and Practice of Constraint Programming*, 2012, Springer Berlin Heidelberg, pp. 118-126.

[16] Nordin, A., Motte, D., Hopf, A., Bjärnemo, R. and Eckhardt, C.-C., 2013, "Constraint-handling techniques for generative product design systems in the mass customization context", *Artificial Intelligence for Engineering Design, Analysis and Manufacturing - AI EDAM*, **27**(4), pp. 387-399.

[17] Nordin, A., Hopf, A., Motte, D., Bjärnemo, R. and Eckhardt, C.-C., 2011, "Using genetic algorithms and Voronoi diagrams in product design", *Journal of Computing and Information Science in Engineering*, **11**(011006).

Paper VIII

Challenges in the industrial implementation of generative design systems: an exploratory study

VIII

Nordin, A.

Artificial Intelligence for Engineering Design, Analysis and Manufacturing

Under review

Challenges in the industrial implementation of generative design systems: an exploratory study

Abstract

The aim of this paper is to investigate the challenges associated with the industrial implementation of generative design tools. Many studies have been aimed at either validating the technical feasibility, or the usefulness of generative design systems, however, there is a lack of research on the practical implementation and adaptation in industry. To that end, this paper presents two case studies conducted while developing design tools for industrial uses. The first case study focuses on an engineering design application and the second case study focuses on an industrial design application. Overall, the results show that the identified challenges are not related to whether the design tools are intended for artistic or technical problems, but rather to the systematization of parts of the design process. The challenges include aspect such as how to fully utilize the potential of generative design tools in a traditional product development process, how to enable designers not familiar with programming to provide design generation logic, and what should be automated and what is better left as a manual task. The paper suggests several strategies for dealing with the identified challenges.

Keywords

Generative design, engineering design, industrial design, design automation, case study

1 Introduction

Generative design systems are generally defined as systems aiming to support human designers and/or automate parts of the design process through computational means (Singh & Gu, 2012), or as Shea et al. state: *"… generative design systems are aimed at creating new design processes that produce spatially novel yet efficient and buildable designs through exploitation of current computing and manufacturing capabilities"* (Shea et al., 2005, p. 263). Generative design systems have been studied since the 1970s. Frazer, for instance, introduced generative system intended for architectural design in 1974 (Frazer, 2002). Since then, there have been a number of attempts to implement generative systems using different approaches for shape generation such as shape grammars, L-systems, cellular automata, genetic algorithms, and swarm intelligence, (e.g., (Singh et al., 2012)) — a comprehensive review of design systems within the realm of evolutionary systems can be found in (Bentley & Corne, 2002). A review of generative design systems applied to industrial design can be found in (Nordin et al., 2013).

Generative design tools not only help the designers enhance their creative repertoire, but can help increase product quality and reduce the number of iterations needed in the product development process through computational means such as interactive feedback or optimization of product performance.

However, many have noted the lack of industrial implementation and adaptation of the design tools developed. Shea et al. for instance noted that *"The real challenge is to make systems that designers want to use in order to investigate the potential for performance-driven generative design to aid negotiation in multi-disciplinary design teams"* (Shea et al., 2005, p. 263). Cagan et al. reasoned that this is because *"… the problem is challenging, a complex balance between representation, generation, and search of a design space in pursuit of original design solutions"* (Cagan et al., 2005, p. 171). Horváth argued that the industry simply does not want to invest in developing premature technology, and would rather hire another designer to help the design process (Horváth, 2005). Another reason for the gap between industry and academia is that the majority of the design tools are developed with academic problems in mind, rather than industrial needs, resulting in tools that might show technical prowess, but lack crucial aspects that hinder their use in an industrial setting. Blessing and Chakrabarti noted that there is, within the engineering design research area, a lack of use of results in practice (2009, pp. 7–8), and that *"…'generic methods' are developed based on the analysis of a specific problem and evaluated using the same problem. In many cases, statements are made about the use of the support, although the evaluation involved only the researcher"* (Blessing et al., 2009, p. 36). There is nothing inherently wrong with developing tools without much industry input, especially since the design problems most commonly targeted for design automation research are in the routine category, rather than the creative design problems (Krish, 2011). The reason for favoring problems with already well-defined product specifications such as product redesigns or benchmark problems is obviously to be able to concentrate on the technical aspects of the design tool rather than development of the product. However, in order to successfully convince industry to use generative tools and to understand what hurdles lie in the way of their implementation, it is necessary to study the whole development process, or as Simon (1973, p. 187) noted: *"there is merit to the claim that much problem solving effort is directed at structuring problems, and only a fraction of it at solving problems once they are structured"*.

Due to the absence of industry input during ideation and development of design tools, and the lack of research on the challenges associated with industrial implementation of generative design systems, it is hard to draw any conclusions on how adaptation could be better facilitated. To this end, this paper presents two exploratory case studies conducted while implementing generative design systems for companies working on product development projects. The aim is to document the challenges associated with the development of generative design systems in practice and to thereby aid the elaboration of strategies for how to overcome them. The first case study focuses on an engineering design application in a company providing solutions for dispensing metal discs. The second case study focuses on an industrial design application in a company working on, amongst other things, surveillance cameras.

2 Research method

2.1 Method

Based on the aim of this paper, and to be able to gain detailed information on the implementation process, a qualitative approach was adopted. This study is exploratory in the sense that it is not intended to test any hypotheses established by the researcher or to refute any existing theory, but rather to collect data on which to generate theory. With these requirements and goals in mind, and based on literature such as (Glaser & Strauss, 1967; Eisenhardt, 1989), it was decided that the best method of achieving them was to perform a number of industrial case studies, during which design tools were developed together with companies. To be able to obtain in-depth first-hand information on the development process, this study is participatory; the researcher is part of the development process and responsible for developing the tools in collaboration with the designers at the companies (Merriam, 1998, p. 101; Blessing et al., 2009, p. 247).

2.2 Case selection

For such an exploratory study, the cases had to be chosen so that they allowed for capturing a large array of data on which to base conclusions. The choice of cases was thus based on theoretical sampling rather than statistical (Glaser et al., 1967; Eisenhardt, 1989). It was decided to find one technology-driven and one user-driven application (as defined in (Ulrich & Eppinger, 2012, pp. 221–222)), as these cases would give more insights on the challenges associated with both engineering design and industrial design. Additionally, it was a requirement that the user-driven application would include both qualitative and quantitative aspects to ensure that the study captured the difficulties in combining the two. After meeting with seven companies, an agreement was reached with two companies for performing the case studies. The first company develops products that are technology-driven, although a large part of the design evaluation is still based on the experience and intuition of the engineering designers. The second company is an industrial design consultancy firm which often works on projects requiring their industrial designers to work with engineering designers. The product the design tool was implemented for was driven by the industrial design aspects, but still needed to satisfy technical constraints.

2.3 Data collection

The data collected during the case studies consists of transcripts of interviews and idea generation sessions, emails, field notes, and time logs of the development work. The interviews were conducted to gain more information on the product and the needs of the designers, as well as getting feedback on the work-in-progress tools. The development was tracked by manually logging the time spent on each activity, as well as logging the save times of all files involved in the project, and then manually tagging each save time with a code representing the type of activity associated with it; for example, updates to the geometry files used were tagged under Geometry, and changes to the source code were tagged as Evaluation function.

2.4 Data analysis

Data analysis consisted in first building a case history detailing the events that had occurred in the two companies, drawing upon the data collected to do so. The case histories were then analyzed for decisions, requests and opinions regarding the generative design system being developed, so as to map key events to a timeline, and to be able to get an overview of the development process. The events were then grouped in categories together with the tags from the time logs, and served as a basis upon which to draw conclusions regarding how different factors affected the overall development process. After this first categorization and analysis had been concluded, the raw data was reviewed once more in order to ensure that nothing had been left out or mislabeled, and to look for similarities and differences between the two cases. The time logs were used to graphically map the key events to the actual development time in order to enable the visualization of the process in a more accessible format.

2.5 Presentation of results

Due to the large amount of data collected during the two case studies, it is only possible to give a summary of the events and conclusions, supported by the quantified results from the time logging. In-depth descriptions of the developed tools have thus been left out in order to be able to concentrate on describing the challenges. Summaries of the case histories are given, as well as a brief description of the developed design tools to enable the reader to put the challenges presented afterwards into context.

3 Implementation

3.1 Structure of the development work

Both case studies were carried out using the same set-up (see Figure 1). During the initial meeting, the company was presented with examples of previous works involving generative design tools, and the general principles, benefits and challenges were discussed. Based on the initial description, the company was then asked to suggest possible applications within their range of products. The needs of the company and the applicability to the research project were then used as criteria for selecting one product to move forward with. The company provided an initial brief of the design problem, for example, constraints, objectives, design space, and so forth. An initial design tool was built based on this brief, in conjunction with interviews and discussions being conducted with the designers. In both cases, the chosen projects were redesign projects, as the companies had previously worked on similar products. The company was presented with the initial design tool early on in the development process in order to better explain the capabilities and workflow of a generative design tool, as neither company had worked with them before. Based on the feedback on the first version of the tool, the second iteration tool was developed, where the majority of the development took place, as the company now had a better idea of what could be done and could propose new functionality or changes to the tool. After the second iteration of the tool had been presented, the goal of the following iterations was to fine-tune the tool until the company was satisfied with its functionality. Each iteration was concluded with a presentation of the tool and a number of product concepts generated by it.

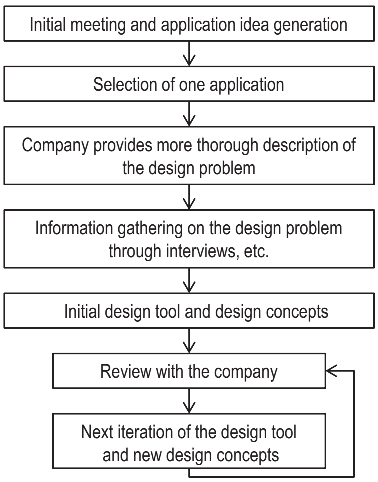


Figure 1. General timeline of the projects

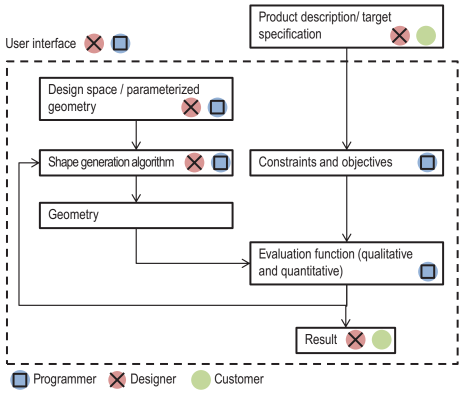


Figure 2. Flow of information for a generic generative design system. The colored dots represent who is responsible for each function. The user interface can include all or some of the functions.

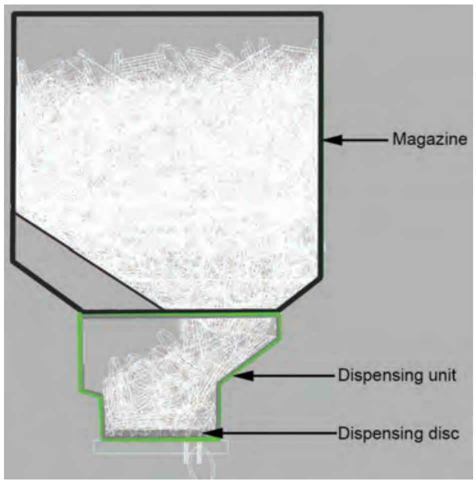


Figure 3. The general layout of the disc magazine and dispensing unit.

3.2 General structure of the tools

Both tools were based on the same generic generative design process, as described in for instance (Frazer, 2002; Cagan et al., 2005; Krish, 2011) and visualized in Figure 2. After the product specifications had been established by the customer and company, they were converted into constraints and objectives by the programmer, and used by the evaluation function to score every design. Each design is based on the design space established by the designer, and the shape generation algorithm formulated by the designer and programmer. After a design has been scored, the results are fed back into the shape generation algorithm to create the next iteration. The results are

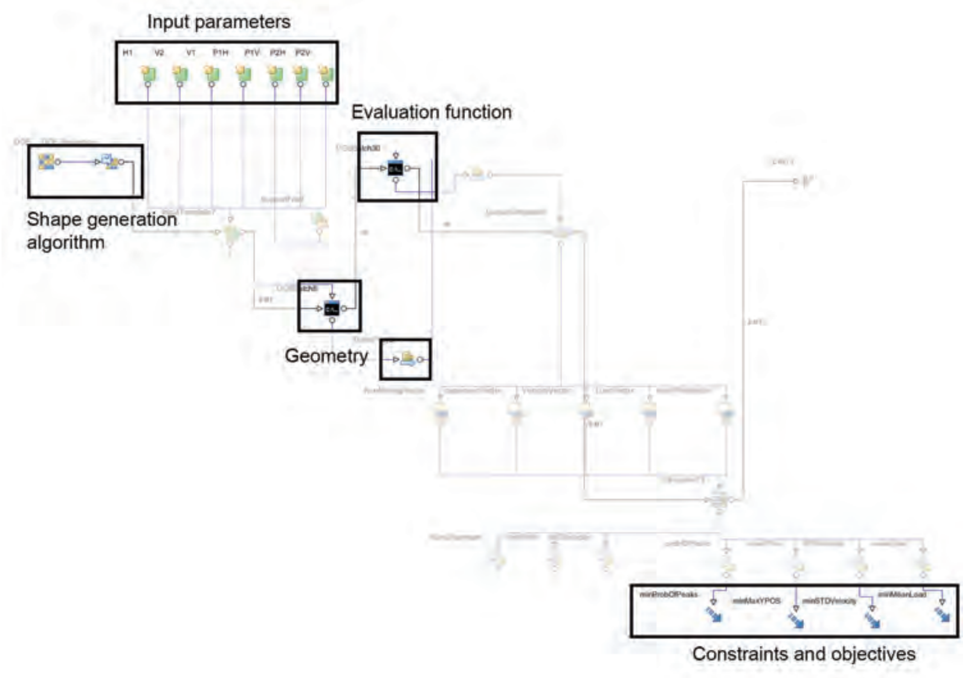


Figure 4. Workflow showing input parameters, constraints and objectives in modeFRONTIER.

monitored and evaluated by the designer and customer until a satisfactory solution has been reached.

3.3 Description of the design problem and design tool development for Company A

Company A is a provider of solutions for dispensing metal discs. Core technologies in their products are the metal disc sorting, storage and dispensing modules.

In this study, the company was interested in in maximizing the metal disc storage and predicting structural overloading of a metal disc dispensing mechanism. The metal disc dispensing system is based on a magazine, which stores the metal discs, and delivers them to the dispensing mechanism, which consists in a wheel with disc sized openings spun by a motor, see Figure 3. The discs fall into the openings and are ejected through a slot which registers them as ejected using a sensor. During normal running conditions, this process is capable of dispensing around 500 discs per minute. The magazine and dispensing mechanism are normally locked and hidden away from the user of the machine, and are not meant to be regularly handled by personnel, thus limiting the need for industrial design of the system.

In an effort to avoid overloading the dispensing units, the company had begun measuring the load on the mechanism, but was not quite sure of how to determine a metric which would signify acceptable or unacceptable loading. Because many of the sub-processes were difficult to observe in real time due to the high rotational velocities, the company also recorded slow motion video of the dispensing process so as to better understand factors that affect the flow of discs and operation of the dispensing mechanism. They had done some statistical analysis of the test data, but it was difficult to base design decisions on it as building new prototypes with modified design features is time consuming and somewhat expensive. Instead, their request was to investigate if simulating the interaction of the magazine and dispensing process was feasible, and to base an optimization on the results.

The problem at hand was to develop this tool while working with the engineering design and testing department of the company, and to establish the actual constraints and objectives, which, for the most part, are not commonly found in classical product development projects. This was beneficial to the case study, as it was not possible to rely on the standard constraints which might apply to most products, such as thermal considerations or structural requirements in terms of stresses, deformations, and so forth. Instead, the company

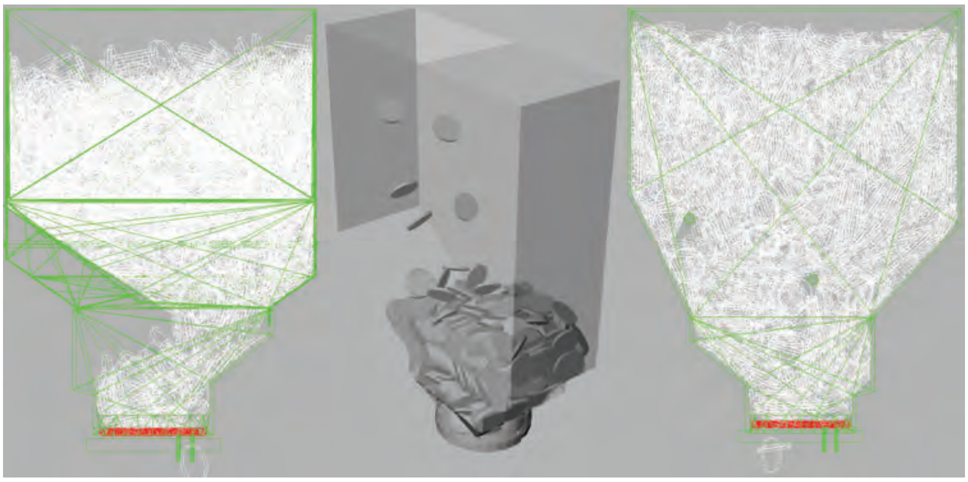


Figure 5. Visualization of the simulation of the disc flow for three different designs.

had so far mostly relied on experience based design rules gained by trial and error.

The development required four iterations of the design tool. The initial iteration of the tool was more of a proof of concept showing that the flow of discs could be simulated with reasonable accuracy and speed. Due to the large amount of rigid bodies, the simulation required around 40 minutes to execute and was thus not interactive. As a result, the tool was more of a classical optimization tool, which searches for the best solutions independently from any interactive user input.

The tool was controlled from Esteco modeFRONTIER (Esteco, 2015) (see Figure 4), which is a general optimization and design of experiments tool, which interacted with PTC Creo Parametric 2.0 (PTC, 2013) and a custom evaluation function implemented using the rigid body simulation library Bullet Physics (Coumans, 2015) (see Figure 5). The output from the tool provided the designer with the number of discs that had been dispensed, the load on the motor, the average velocity of the discs, and the height of the highest disc as functions of time in intervals of .01 s. Each optimization, consisting on average of 320 design evaluations, required around 24 h running on eight 2.4 GHz Intel Xeon E5620 cores.

The company had not previously been able to establish any performance metrics for the product, and as such a large part of the development was concerned with finding good metrics which could be simulated. In the final tool, the designer could either use modeFRONTIER to specify parameter intervals and objective functions and run an optimization or design of experiments using the built-in functionality, or they could access the evaluation function directly via Creo Parametric to evaluate an arbitrary geometry and set the number of discs to be simulated and the simulation duration. It also provided the designer with a 3D view of the flow of discs, that is, even if no metric could have been established, the designer could have acted as the evaluation function in much the same way as the company had evaluated design concepts previously.

3.4 Description of the design problem and design tool development for Company B

Company B is an industrial design consultancy firm which employs industrial and engineering designers. Their clients are mainly companies developing consumer electronics such as cameras, cell phones and health-care products.

The project was based on one of the reoccurring client’s products, a camera, in which the heat dissipation mechanism had previously been hidden, or at least not prominently featured. The firm has had several projects in which heat dissipation was important for the function of the products. The company is often hired to design several similar variants of the same product typology, such as a small affordable version for home use, a medium version, and a high-end professional version, all requiring different thermal and case designs but keeping the same general design language. This has led the company to become interested in being able to generate designs that conform to the customer’s brand or design language, while also satisfying technical constraints such as case temperature, and leaving the designer to explore several ideas without being tied down by technical analysis. As more powerful components were integrated into the camera, due to increased competition in the market, the amount of heat that needed to be dissipated had increased to a point where passive cooling could no longer be easily hidden, or a switch to forced convection would have to be made, which is not ideal in many cases. Instead of ruining the

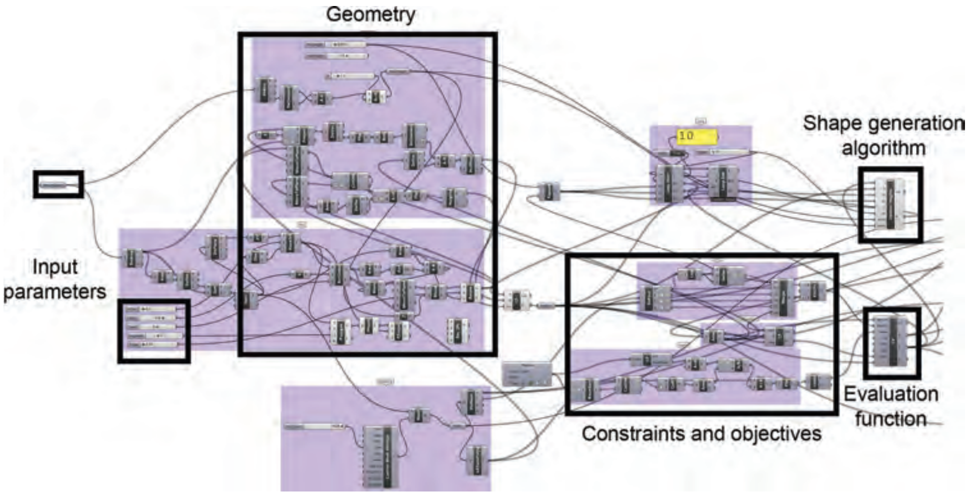


Figure 6. Workflow of the last iteration tool in Grasshopper showing the visual programming language.

aesthetic of the product by placing generic heatsinks on the outside shape of the product, it would be preferable to have a custom heat dissipation design which fits with the overall product expression and company brand. However, the design firm does not have in-house expertise within thermodynamics, and has thus so far been relying on their customers' experts to get feedback on their design concepts. This has, in their opinion, led to final designs that are not as innovative as they could have been had the exchange of concept ideas and feedback on them been faster. The task was thus to work with the industrial designers of Company B to develop a design tool which could take into account the thermal constraints of the product, while leaving the designer to control the industrial design aspects.

In comparison to the application for Company A, this application revolves around the qualitative aspects, and thus requires a different type of tool in which the designer is more involved with the development of the shape generation algorithm and overall interface of the tool.

The tool required four iterations before satisfying the needs of the designer and the company. The initial iteration of the tool implemented a rudimentary custom FEM-based evaluation function, enabled the designer to change the geometry of the heatsink, change the position and values of the thermal loads, and to evaluate the temperature distribution on the body of the product. The second iteration of the tool implemented several shape generation algorithms from which the designer could choose. The remaining iterations were spent on refining the shape generation algorithm and improving the ease of use with which the designer could define the design space.

The tool was developed for use with the CAD tool Rhinoceros 3D (Robert McNeel & Associates, 2015), using a plug-in for visual programming, Grasshopper (Robert McNeel & Associates, 2014), with custom thermal simulation code written in C# (see Figure 6). The tool itself was interactive; the update rate of the shape generation and feedback was about 2 Hz on a four core Intel Core i7 950 3.07 GHz processor.

The shape generation algorithms were based on 1) increasing the thickness of the heatsink in proportion to the temperature (proposed by the industrial designer), 2) modulating the height of the fins based on the temperature and 3) basing the thickness of the heatsink to maximize the heat dissipation. In the end, the industrial designer could simply specify a 3D volume as the design envelope, specify the thermal loading using 2D curves and power ratings, fix certain areas of the design space to for instance define ribs in the heat sink (see Figure 8), and get visual and numeric feedback of the temperature gradient (see Figure 9 and Figure 7). The designer could choose to manually define the heat dissipation shape, or to let the algorithm automatically define the shape based on the temperature results over a number of iterations. The tool was applied to a range of camera variants to ensure that the tool could handle different design spaces without requiring substantial modifications.

Figure 7. Examples of the output from three shape generation algorithms. Left: maximize thermal dissipation; middle: increase thickness of heatsink in relation to the local temperature; right: user-definable heights in combination with temperature modulation of the remaining geometry.

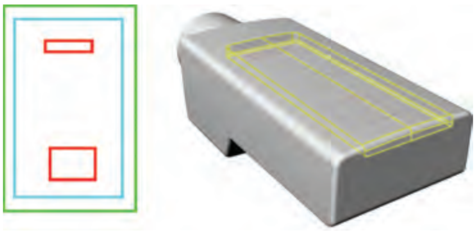


Figure 8. Left: the inputs to the first iteration of the tool with thermal loads in red, design space in blue, and convection area in green. Right: the inputs to the final iteration of the tool with the design space represented by a 3D volume in yellow, and the thermal loads in red.

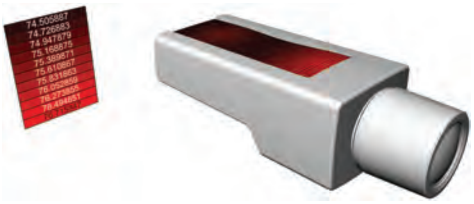


Figure 9. The feedback from the final iteration of the tool showing the temperature distribution and adapted shape of the heatsink.

3.5 Timeline of the development projects

The project with Company A was conducted over 83 days, although the development time was concentrated to a few periods of activity due to time constraints on behalf of the company. In total, the time spent on active development was 105 h. A summary of the development time is shown in Table 1. During the development, five team meetings and four one-on-one interviews, averaging 40 minutes in length, were conducted. A summary of the development timeline is shown in Figure 10.

The project with Company B was conducted during 71 days. In total, the time spent on active development was 58.5 h, considerably less than the development time for Company A. A summary of the development time is shown in Table 2. During the development, three team meetings and four one-on-one interviews, averaging 60 minutes in length were conducted. A summary of the development timeline is shown in Figure 11.

To differentiate between the causes for iterations, the development time is split into two sections. The first part is up until the first iteration of the tool has been presented (the first iteration of the tool is to demonstrate what is possible based on the initial brief). The second part is until the design tool is finished, that is, the company is satisfied with it. In Company A, about 45% of the development time was spent on updating the tool based on the designer's inputs. In Company B, the time spent was 42%. In Company A, the major part of the development time was spent on developing the evaluation function, as there were no existing ways of evaluating the product. However, most of this time was spent during the initial phase of the project and did not require much input from the company. The second and third most time-consuming activities were development and testing of constraints and objectives, and the adaptation and parameterization of geometry to the evaluation system. These two activities represent the constraints, objectives and design space of the problem, which are often assumed to be given, but in actuality represent almost half of the development time, even though the product is not a new product for the company but rather a re-design. The optimization time has not been included in the data in Table 1 as it is not active development time. In Company B, the most time-consuming activity was again the development of the thermal evaluation function. The second most time-consuming activity was the development of the shape generation algorithm, where several different approaches were tested for how the shape of the product should respond to the results of the thermal simulation. Included in this time is also the development of the user interface where the designer could control the design space and shape generation parameters. In the case of Company B, the geometry parameterization and adaptation did not require much time. The reasons for this was firstly, that the only parameter was the height map of the product shape, and secondly, that the company could provide sufficiently simple geometry to be used directly with the design tool.

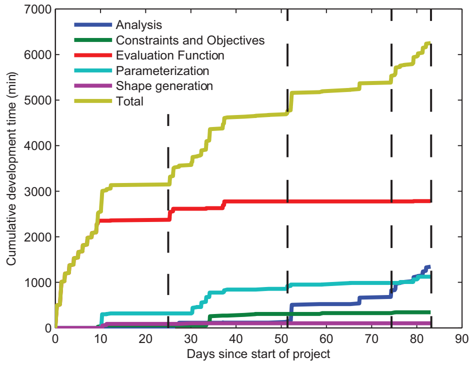


Figure 10. Timeline of the development of the tool for Company A over 83 days. Each vertical line represents one review meeting.

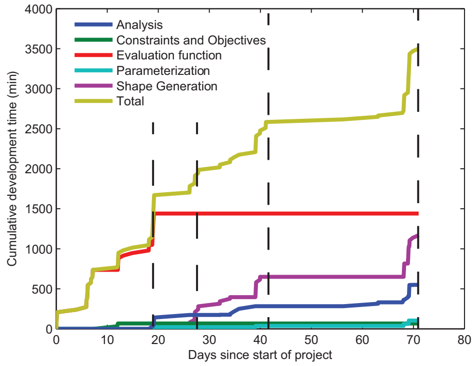


Figure 11. Timeline of the development of the tool for Company B over 71 days. Each vertical line represents one review meeting.

Table 1. Summary of the time spent on each development activity in Company A.

Analysis of Output and Testing	22.5 h
Constraints and Objectives	11 h
Development of Evaluation Function	46.5 h
Parameterization	19 h
Development of Shape Generation Algorithm	6 h
Total time	105 h
Time spent after first iteration	47.5 h

Table 2. Summary of the time spent on each development activity in Company B.

Analysis of Output and Testing	12 h
Constraints and Objectives	1 h
Development of Evaluation Function	24 h
Parameterization	2 h
Development of Shape Generation Algorithm	19.5 h
Total time	58.5 h
Time spent after first iteration	24.5 h

4 Challenges encountered with commentary and recommendations for future projects

In this section, a number of challenges observed during the development of the two design tools are described. The first three challenges relate to the overall development of a generative design system, and the attitudes towards it among the two companies. The next two challenges are in relation to the definition of the constraints and objectives of the generative design system. Finally, the last challenge describes a more practical implementation issue relating to geometry and parameterization.

In the following subsections, each challenge will first be described in a general manner; the challenge will then be illustrated through examples from the case histories; and finally, the strategy adopted in the case study to deal with the challenge will be described and discussed.

4.1 Moving from automation to generation

It can be difficult for a company that has never dealt with generative design tools before to fully grasp the possibilities and utilize them in an optimal manner. Although there are many benefits with a design tool tailored to the design process, it requires a substantial investment of resources that could potentially be better spent elsewhere. It is therefore of importance that the application is selected with the unique capabilities of generative design systems in mind.

During the initial meeting with Company B, the overall attitude towards generative design was positive, and, in the following idea-generation meeting, the designers were eager to suggest uses within their own projects. Most proposed applications were either purely technical, concentrating on aspects of a product which, although crucial to the performance, were not part of the overall product expression, or purely aesthetic, concentrating on the possibility of generating complex shapes that were too time-consuming to achieve manually. Few suggestions included both qualitative and quantitative considerations. The reason of the lack of applications where the aesthetic and functional parts are combined is probably based in the fact that the more technical parts of a product, such as heatsinks or ribbing on plastic details, have traditionally been constructed by engineers purely for function and then hidden away by the industrial design team. Another aspect, which was discussed by the company, was the potential in being able to use the fact that the shape was algorithmically generated and optimized as a sales argument. Not because the product necessarily performed any better, but because the story behind the design process could be enough to make it more attractive than other offers on the market.

The strategy adopted in the case of Company B, to push them towards applications more in line with the capabilities of generative design systems, was to discuss what possibilities could arise if the technical parts become an integral part of the product expression. This is perhaps something they have tended to avoid because of the extra complexity, but by freely generating ideas, without considering the feasibility of implementing them, a number of new applications could be found, amongst them the application described in this paper.

4.2 Moving from designing a product to designing an algorithm

In the ideal case, the designer using a generative design system is also the programmer. However, even in that scenario, designing an algorithm instead of an object is a challenging task, or as Knuth puts it regarding the design of a shape generation algorithm for fonts, *"Meta-design is much more difficult than design; it's easier to draw something than to explain how to draw it*" (Knuth, 1995, p. 1).

In Company B, the designer would typically describe how they wished the output of the tool would look, rather than suggest new logic for the algorithm when suggesting modifications to the shape generation algorithm. To deal with the challenge of the designer not directly being able to input new shape generation logic, a number of methods for finding a satisfactory shape generation algorithm were used. The first was based on emulating the design reasoning the designers had employed when working manually. This led to a shape that the designers were satisfied with, but it left little in terms of control over the shape. This morphology was then expanded upon by allowing the designer to lock certain aspects of the design space, in this case the heights of user-defined areas acting as ribs for a heatsink. A third option was presented which was based on the thermal performance of the heatsink rather than the aesthetics. In this case, although the thermal performance was of importance, the driving factor behind the design decisions made by the designers was the aesthetic of the shape and how well it fit the company's brand, rather than how well it dissipated heat.

Letting the designer describe the shape generation logic to the programmer, in a way, moves the bottleneck from the designer interacting with the engineers who evaluate the technical feasibility of their concepts, to interacting with a programmer. However, a large part of the designer's task is to continuously evaluate the form based on a set of criteria and being able to justify their design decisions, which is conceptually not very different from defining the logic an algorithm should follow to create the form of a product. Another aspect is that the quality of the design tool will most likely be improved by including the end-user in the development process. Janssen (2006) for instance notes the importance of involving the designer in the software creation process as it creates a positive feedback-loop between the system and designer, that is, tools are better co-developed with the designer with a design task at hand. However, this approach still lacks many of the benefits that generative design has to offer, where the creator of the system can experiment with different shape generation logic without the burden of manually having to construct the oftentimes complex geometry, and the serendipitous emergence of shape found in, for instance, (Sims, 1991).

4.3 Knowing what to automate

As previously noted in (Nordin et al., 2010) regarding the balance between control and automation, a compromise in level of user control most often leads to unsatisfactory results. The study showed that the industrial designers tended to want to have as much control as possible over the shape, while still whishing that the tedious and laborious parts be automated, or towards letting the design tool completely determine the design. If the design process is rather repetitive, automating it is straightforward. However, in the case of most generative systems, the point of the tool is not merely to perform repetitive tasks quickly, but also to intelligently couple the shape of the product to the performance metrics set by the designer. Developing an algorithm for doing design tasks that are almost solely based on the designer's subjective opinion can be time-consuming as the designer is refining his or her vision of the product based on the output of the algorithm. Moreover, a more specialized tool will not be as applicable to other products, which could have made the initial investment more economically justified.

In the case of the heat dissipation problem in Company B, it was for instance requested that the algorithm would more closely match the curvature of the surrounding shape, and that the resulting mesh would be automatically post-processed into a smoothened NURBS-surface. It was however difficult to know how exactly the designers whished that these operations should be performed, without performing several iterations of the algorithm.

The strategy adopted in the case of Company B was to not aim for an all-encompassing tool that automates the entire workflow, but rather a tool that acts as an inspiration and concept generator to the designer. If need be, the process of transforming the output of the generative design system to something that can be input into a computer aided manufacturing tool can later be automated in a separate tool, thereby retaining the genericity of the generative design system and reducing the development time.

4.4 Replacing rules-of-thumb with measurable constraints and objectives

In order for the optimization or automatic generation of a product to be possible, the objectives and constraints associated with the product must be possible to measure, either through virtual or physical tests, or through user feed-back. The problem of finding suitable metrics is not unique to generative systems, the general recommendation in for instance Ulrich and Eppinger (2012) and Ullman (1997) is for the product specification to be based on measurable metrics and target values. However, in practice this might not always be strictly followed since the company might not think the investment in determining metrics and developing methods for evaluating them is worthwhile compared to simply basing the evaluation on trial and error or the experience and intuition of the designers.

In the case of Company A, the product was not subject to any substantial qualitative requirements, and as such there was no need for user involvement. Instead, the focus was on replacing the rules-of-thumb with metrics that could be measured by means of simulation of the flow of discs. This involved two major challenges: first, the design rules themselves were not thoroughly documented, but rather existed as part of the expertise of the designers; second, the rules were often based on experience, rather than something which was measurable, and the rationale behind the rules needed to be found out, or at least the reason they were put in place to prevent or improve. An example of a rule-of-thumb, which was discovered through interviews, was that the designer preferred to keep internal planes asymmetric. The rationale behind this was that this practice could decrease the risk of bridge building; however, they had no way of directly measuring the frequency of bridge building in their physical prototypes. Because of this, a metric needed to be formulated based on the physical characteristics of a disc bridge. The first proposed metric was based on measuring the rate of discs being dispensed, but, after discussions with the designers, this metric was deemed inadequate since bridges could still form higher up in the magazine, without affecting the flow of discs further down in the stack. Another metric was conceived, based on measuring the standard deviation of the mean velocity of all the discs. A high standard deviation indicated that discs tended to get stuck and suddenly drop as bridges were formed and dissolved.

Generally, the rules first needed to be found out through interviews. Based on the aspect of the product's performance the design rule was put in place to improve, a metric needed to be formulated based on what was conceivably possible to measure through simulation of the disc flow. The metrics then needed to be implemented in the evaluation function, and fine-tuned, which is an iterative process as the output from the generative design system must be evaluated by the designers in order to determine whether or not the constraints and objectives lead to feasible designs. Furthermore, Company A had previously successfully developed similar products, indicating that they had an in-depth understanding of the constraints and objectives of the product, but expressing them in measurable metrics still proved to be challenging.

4.5 Avoiding loopholes in the constraint and objective formulations

When management formulates a specification for use by a designer, there is a certain amount of shared knowledge and experience which is assumed to be shared between the two parts. It is assumed that the designer will use his or her best judgement to not propose nonsensical solutions, even though they might formally fulfil the design specification. In the case of optimization algorithms, this, however, is not the case (e.g., (Thompson, 1997)). If there is an easily found way of achieving good results, the algorithm will tend towards that part of the search space, even if the solutions are obviously unfeasible to a human designer.

Examples of such loopholes encountered during the development of the design tools are unexpected regeneration errors in the CAD-systems leading to undefined behavior. For instance, if the geometry of the disc magazine is corrupted in such a way that the discs can easily escape it without passing through the intended dispensing mechanism, the evaluation function will likely favor it over valid geometries, leading the optimization to focus on an area of the design space which leads to corrupted geometry. In other cases, the internal planes could obstruct the flow of discs in such a way that the metrics based on motor load and standard deviation of the disc velocity scored high, but the other metrics scored poorly. These design points nevertheless represented Pareto-optimal points, and the design space was unnecessarily examined by the algorithm.

Recommendations to avoid problems with the geometry is to always verify the design space before launching the optimization in order to quickly find obvious problems, for instance by running a large DOE just on the geometry, and to include sanity-checks in the constraints, such as the number of discs dispensed must be lower than the total number in the magazine. However, although many of these problems can be minimized by careful examination of the design space and geometry definition, there always remains a risk that some unforeseen issue remains, and one needs to take this into account when estimating the development time.

4.6 Parameterizing and simplifying geometry

In any computer based simulation, the preprocessing of the geometry accounts for a large part of the engineering time. Decisions include what details are important to the results and how to best discretize it if for instance a finite element analysis is to be performed. Additionally, if designs are to be automatically generated based on the results of the simulation, there is an added layer of complexity in parameterizing the geometry.

As can be seen in the summary of the development time in Company A, adapting geometry to fit the design tool and evaluation function took up a considerable part of the time. They had not previously dealt with any computer-based simulation of this part of their product, and as such had no CAD-files suitable for simulation or optimization. The major difficulty in simplifying the design, especially as an outside party with no former experience of the design rationale behind the products, was to interpret which parts could be removed from the geometry, and how the design was allowed to change without interfering with other parts of the product. .

The task of simplifying and parameterizing the geometry could have been left to the company; however, the problem on their end would be to understand the intricacies of design parameterization and what geometrical features are computationally expensive or unsuitable to include. In either case, it should be expected that a major part of the development time will be spent on adapting and creating geometry if the shape generation or optimization is not applied to a relatively isolated part of the product, as in the case of Company B.

5 Conclusion

The results of this study details the challenges that were encountered during the development of two generative design systems intended for industrial applications. The two case studies were based on one technology-driven application in Company A and one user-driven application in Company B. The first observation regarding the differences between the two was that the changes to the design tool requested by the Company A were oriented towards adding new measurements, changing the objectives of the optimization and including more phenomena in the simulation. The requests put forward by Company B were entirely focused on the shape generation. This might not be especially surprising, but it shows that many decisions regarding aspects less important to the company will lie in the hands of the developer. Another difference in attitude towards design tools was observed in terms of who would be the end user. Company A showed reluctance towards using the developed tools on their own; they would instead prefer to consult the programmer when new designs needed to be generated. Company B was more inclined towards letting their industrial designers use the tool to generate new designs. This could be explained by the difference in emphasis on qualitative and quantitative aspects of the design. It also shows that it is important to decide who will be the user before the development starts, as very different requirements are put on an interface intended to be used by someone not familiar with the design tool. Overall, the challenges identified are not related to whether the design problems are artistic or technical in nature, but rather to the systematization of parts of the design process.

The list of challenges found in this study is by no means exhaustive, but rather scratches the surface of the implementation issues that might be faced when developing generative design systems. Issues relating to the integration of generative design systems into the company's organization, data management system and development routines need to be studied before a fully mature system can be achieved. Additionally, aspects relating to any commercial software needs to be considered, such as how maintenance, licensing and reliability should be handled. However, the study does offer an in-depth view into a number of hurdles that most likely will be encountered in similar projects, and since, as previously noted, there is a lack of similar studies of industrial projects, this paper also serves as a starting point for further investigation and future recommendations or methodologies for how to make the processes more efficient.

6 References

Bentley, P. J., & Corne, D. W. (2002). An introduction to creative evolutionary systems. In P. J. Bentley & D. W. Corne (Eds.), *Creative Evolutionary Systems* (pp. 1–75). San Francisco, CA: Morgan Kaufmann. http://doi.org/10.1016/B978-155860673-9/50035-5

Blessing, L. T. M., & Chakrabarti, A. (2009). *DRM, a Design Research Methodology*. London: Springer. http://doi.org/10.1007/978-1-84882-587-1

Cagan, J., Campbell, M. I., Finger, S., & Tomiyama, T. (2005). A framework for computational design synthesis: Model and applications. *Journal of Computing and Information Science in Engineering*, *5*(3), 171–181. http://doi.org/10.1115/1.2013289

Coumans, E. (2015). Bullet Physics Library. Retrieved August 6, 2015, from http://bulletphysics.org/

Eisenhardt, K. M. (1989). Building Theories from Case Study Research. *Academy of Management Review*, *14*(4), 532–550. http://doi.org/10.5465/AMR.1989.4308385

Esteco. (2015). modeFRONTIER 2014. http://www.esteco.com/modefrontier.

Frazer, J. (2002). Creative design and the generative evolutionary paradigm. In P. J. Bentley & D. W. Corne (Eds.), *Creative Evolutionary Systems* (pp. 253–274). San Francisco, CA: Morgan Kaufmann.

Glaser, B. G., & Strauss, A. L. (1967). *The Discovery of Grounded Theory: Strategies for Qualitative Research*. *Observations* (Vol. 1). New Jersey, NY: Transaction Publishers. http://doi.org/10.2307/2575405

Horváth, I. (2005). On Some Crucial Issues of Computer Support of Conceptual Design. In D. Talabă & T. Roche (Eds.), *Product Engineering: Eco-Design, Technologies and Green Energy* (pp. 123–142). Dordrecht: Springer.

Janssen, P. (2006). A generative evolutionary design method. *Digital Creativity*, *17*(1), 49–63. http://doi.org/10.1080/14626260600665736

Knuth, D. E. (1995). *The METAFONT book* (2nd edition). Reading, MA: Addison–Wesley.

Krish, S. (2011). A practical generative design method. *Computer-Aided Design*, *43*(1), 88–100. http://doi.org/10.1016/j.cad.2010.09.009

Merriam, S. B. (1998). *Qualitative research and case study applications in education* (2nd edition). San Francisco, CA: Jossey-Bass.

Nordin, A., Motte, D., Hopf, A., Bjärnemo, R., & Eckhardt, C.-C. (2010). Complex product form generation in industrial design: A bookshelf based on Voronoi diagrams. In J. S. Gero (Ed.), *4th Design Computing and Cognition Conference - DCC'10* (Stuttgart, pp. 701–720). Dordrecht: Springer.

Nordin, A., Motte, D., Hopf, A., Bjärnemo, R., & Eckhardt, C.-C. (2013). Constraint-handling techniques for generative product design systems in the mass customization context. *Artificial Intelligence for Engineering Design, Analysis and Manufacturing - AI EDAM*, *27*(4), 387–399.

PTC. (2013). Creo 2.0 [Software]. http://www.ptc.com/cad/3d-cad/creo-parametric.

Robert McNeel & Associates. (2014). Grasshopper [Software]. http://www.grasshopper3d.com/.

Robert McNeel & Associates. (2015). Rhinoceros 3D [Software]. http://www.rhino3d.com/.

Shea, K., Aish, R., & Gourtovaia, M. (2005). Towards integrated performance-driven generative design tools. *Automation in Construction*, *14*(2), 253–264. http://doi.org/10.1016/j.autcon.2004.07.002

Simon, H. A. (1973). The structure of ill structured problems. *Artificial Intelligence*, *4*(3-4), 181–201. http://doi.org/10.1016/0004-3702(73)90011-8

Sims, K. (1991). Artificial evolution for computer graphics. In T. W. Sedeberg (Ed.), *SIGGRAPH '91* (Las Vegas, pp. 319–328). New York, NY: Association for Computing Machinery.

Singh, V., & Gu, N. (2012). Towards an integrated generative design framework. *Design Studies*, *33*(2), 185–207. http://doi.org/10.1016/j.destud.2011.06.001

Thompson, A. (1997). An evolved circuit, intrinsic in silicon, entwined with physics. In T. Higuchi, M. Iwata, & W. Liu (Eds.), *Evolvable Systems: From Biology to Hardware - First International Conference*, *ICES96* (Tsukuba, pp. 390–405). Berlin Heidelberg: Springer. http://doi.org/10.1007/3-540-63173-9_61

Ullman, D. G. (1997). *The Mechanical Design Process* (2nd edition). Singapore: McGraw-Hill.

Ulrich, K. T., & Eppinger, S. D. (2012). *Product Design and Development* (5th edition). London: McGraw-Hill.