DECOMPOSITION-BASED ASSEMBLY SYNTHESIS OF FAMILY OF STRUCTURES

by

Onur L. Cetin

A dissertation submitted in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy
(Mechanical Engineering)
in The University of Michigan
2003

Doctoral Committee:

Assistant Professor Kazuhiro Saitou, Chair
Professor Shixin Jack Hu
Associate Professor Jianjun Shi
Adjunct Assistant Professor Donald E. Malen

ACKNOWLEDGMENTS

I would like to first of all thank my advisor Prof. Kazuhiro Saitou for his constant support and guidance throughout this study.

Cordial thanks to all dissertation committee members for their guidance, and specifically to Dr. Donald Malen for his kind help in developing the joint library efficiently used in this project.

I do appreciate the extensive work by my office mates Byungwoo Lee and Karim Hamza for the visualization of the results using ACIS and development of the multi-objective genetic algorithm code, respectively. Toyota Central R&D's contributions are also acknowledged, for providing several realistic design problems throughout the evolution of the assembly synthesis method.

This work has been partially supported by the National Science Foundation under CAREER Award (DMI-9984606), the Horace H. Rackham School of Graduate Studies at the University of Michigan, and General Motors Corporation through General Motors Collaborative Research Laboratory at the University of Michigan. These sources of support are gratefully acknowledged.

Finally, many thanks to my family and friends (especially to Cem Baydar and Emre Kazancioglu) for their ongoing, invaluable support.

TABLE OF CONTENTS

CHAPTER 1

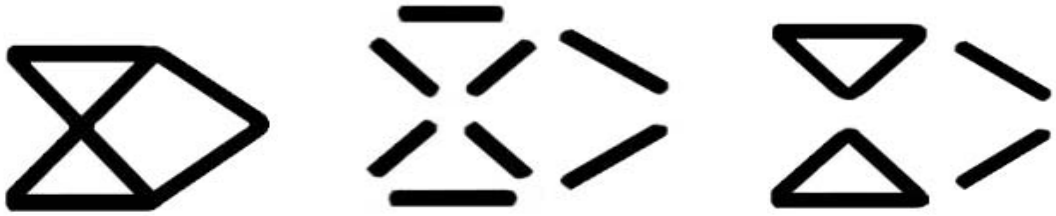
INTRODUCTION

1.1 Problem Statement and Objectives

Mechanical products are very rarely monolithic; one of the reasons is that assembly of components allows simpler forms for the individual components, which are often more inexpensive to manufacture (Gupta *et al.*, 1997). On the other hand, Design for Assembly (DFA) methodologies (Boothroyd *et al.*, 1994) often suggest the reduction of the number of components and joints to minimize the assembly cost. Furthermore, the structural products usually favor fewer joints, since very often joints are the weakest points: for instance many fatigue failures are initiated from welded joints. Therefore, the question is: "assuming a joint has to be made, what is the best method to do it?" (Le Bacq *et al.*, 2002).

Recognizing that the decisions on *where and how* the joints are to be made heavily impact the subsequent design stages of individual components, *decomposition-based assembly synthesis* method was developed (Yetis and Saitou, 2000; Yetis and Saitou, 2002) for the early identification of the joint locations and designs that minimally impact the overall structural strength. In this method, the designer defines all possible joint locations and provides feasible types of joints at each location (including a type for "no joint"), among which the optimal selection can be made. This systematic approach aims to explore a large number of decompositions prior to the detailed component design phase, also integrating manufacturability and assemblability guidelines to guarantee that fabrication of the resulting components is feasible. A demonstrative example is given in

Figure 1.1. As illustrated in in Figure 1.1(b), in this example potential locations of joints are defined to be the end of each beam element. The method allows alternative definitions to restrict the placement of joints or to identify the simplest components in each structure.



(a) (b) (c)

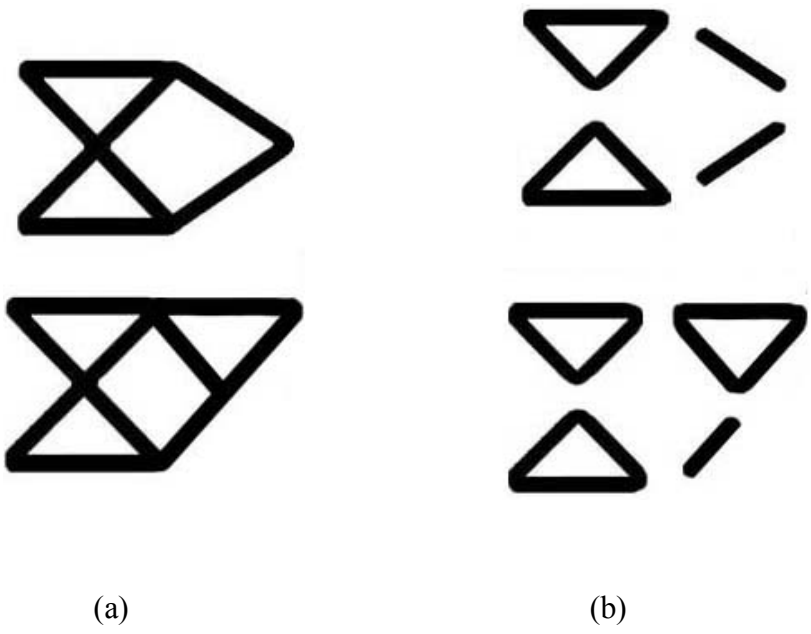
Figure 1.1. (a) A 2-D structure to be fabricated, (b) The case of maximum decomposition, when the structure is to be produced by welding the simplest possible components together, (c) Optimal decomposition as a result of decomposition-based assembly synthesis method.

Modular product design – sharing components across multiple products – is viewed as a convenient way to offer high product variety with low production cost. The basic premise here is that the component sharing would result in less design effort and fewer production varieties with higher volumes, hence reducing overall production cost.

The main objective in this thesis is development of a method to identify sharable components early in the design process. Recognizing that this problem pertains to the conceptual design stage at which the decomposition of the product is carried out, the approach in this study will be evolving the assembly synthesis method to incorporate the design for modularity concept.

When using the decomposition-based assembly synthesis for a family of products, based on given criteria, two or more structural products can be simultaneously decomposed by optimally selecting the locations and types of joints. During the process the decomposed components can be compared with respect to geometric similarity and

potential shared modules can be easily identified. Interfaces of modules can be also forced to be made identical, a natural requirement to be able to use the same component for a family of products. The structures are assumed to bear some similarity but can be distinct in the geometry and/or loading conditions. Figure 1.2(a) shows an example of two such variant structures, and Figure 1.2(b) gives the results of the assembly synthesis when applied to two products simultaneously, with a constraint that enforces modularity. The decomposed triangular components are found to be sharable components between the structures.



(a) (b)

Figure 1.2. (a) Two 2D structures to be fabricated and simultaneously given as an input to decomposition-based assembly synthesis, (b) Optimal decomposition that leads to the minimum reduction in structural strength while sharing some components (modules).

Decomposition-based assembly synthesis of a family of structures is posed as an optimization to minimize the reduction of structural strength due to the introduction of joints, while maximizing the manufacturability, assemblability and component sharing between two structures. The method is to be applicable to both 2D and 3D structural products and the optimization process is aimed to be efficient enough to handle structures

as complex as automotive body-in-white models. It is believed that the following critical questions have to be answered to bring about a satisfactory solution to this problem:

1. An essential research issue is realistic modeling of joints for an improved joint design process. How can human designers' experience be incorporated in the process to guarantee that for each structure a feasible set of joint alternatives are available?
2. It is considered another milestone to utilize objective function terms and constraints that fully represent the assembly and manufacturing processes, so that potential problems in future design steps can be solved in advance. How should the formulation be modified to fully capture the dynamics of the production process?
3. It is usually presumed that design for modularity is beneficial due to potential cost reduction; how can this fact be proved in a quantitative way, rather than taking the benefit of component sharing as a premise?
4. Assembly synthesis is inevitably a multi-objective problem and a conceptual design tool should be capable of exploring the trade-offs related to the given objective functions. How can a set of alternative solutions be generated, considering varying degrees of influence of the design criteria?
5. It is imperative that an efficient design tool at conceptual design stage is easy-to-use and is able to give quick but reliable results. What should be the level of human input during the process? Which optimization techniques should be used for best performance?

1.2 Motivation

As the global competition increases rapidly, manufacturing industry struggles to bring well-designed and well-manufactured products to market in a timely fashion. Although product design incurs only a small fraction of the total product cost, the

decisions made during the design phase account for a significant portion of this cost and prove crucial to the success or failure of the product. The time and cost involved in making engineering changes, in-process adjustments and the like increase rapidly as the product development process evolves. Early anticipation and avoidance of manufacturing and assembly problems can have a huge impact in reducing the product development time (Gupta *et al.*, 1997; Mantripragada and Whitney, 1998).

Increasing research attention is being directed toward the integration of engineering design and manufacturing to achieve the efficient and timely product development. These attempts have led to the evolution of design for manufacturing (DFM) and design for assembly (DFA) methodologies. These involve simultaneously considering design goals and manufacturing and assembly constraints in order to identify and alleviate problems while product design; thereby reducing the lead time for product development and improving product quality. Most existing approaches generate redesign suggestions as changes to individual feature parameters, but because of the interactions among various portions of the design, it is often desirable to propose a judiciously chosen combination of modifications (Gupta *et al.*, 1997).

Modularity is a tested and proven strategy in product design. One short description would be having products with identical internal interfaces between components. The scope of the word *interface* includes the connection between components in functional, technology and physical domains. The interfaces between components are seen by many as the core issue of modularity and they must be standardized to allow the ability of the full exchange of components (Blackenfelt and Stake, 1998).

Design for modularity is now in widespread use globally. Carmakers prefer designing many features of a family of cars at the same time, instead of one model at a time. Standardizing components and letting several variant products share these components would save tooling costs and many related expenses (Sundgren, 1999; Kota

et al., 2000; Muffatto and Roveda, 2000). Developing a complex product involves many activities and people over a long period of time. Making use of modularity leads to the clustering of activities involved in the design process, so the potential group of activities might be scheduled simultaneously, which enables simplification of project scheduling and management (Blackenfelt and Stake, 1998). As the identification of modules tremendously affects the entire product development process, the strategy is usually applicable in the early phases of the design process.

During conceptual design, teams of designers generally begin to develop a new product by sketching its general shape on paper. This "back of the envelope" approach is key aspect of the creative thought process. Often, if the manufacturing engineers noticed any manufacturing-related problems while examining the blueprints of the product design, they would notify the design team and the design would be sent through another iteration. To expedite these time-consuming iterations, as a tool for the engineers during the brain-storming period, this project aims to achieve a systematic decomposition process to carry out assembly synthesis, combining the DFA and DFM concepts with the design for modularity process.

A major motivation for this thesis is the fact that assembly synthesis, as well as joint design in general, is overlooked in the structural optimization literature. Probably it is a consequence of this fact that design for modularity, a concept that naturally fits into the decomposition process, has not been applied to structural products. Therefore, a novel, efficient technique is needed to encapsulate these ideas in a multi-objective design optimization context, to fully explore the trade-offs in the complex problem of product family design.

1.3 Outline of the Thesis

The thesis is organized as follows: Chapter 2 gives a literature survey on several concepts introduced in this chapter, including integration of DFA and DFM into automated design, design for modularity/product platforms and related cost modeling approaches. Chapter 3 presents the application of the assembly synthesis for 2D continuum-based structures, taking images of structural product configurations as inputs. Chapter 4 extends the analysis to 2D and 3D beam-based product models, increasing the applicability of the method in real-life: a case study on automotive body structures is given at the end of this chapter. Chapter 5 adapts the formulation to aluminum space frames (ASF), updates the method for more efficient optimization process and demonstrates the results with a case study based on two ASF based real automotive bodies. Chapter 6 concludes the thesis, discussing the capabilities and limits of the method as well as the possible future research.

This outline is constructed to reflect the evolution of the decomposition-based assembly synthesis; even though the different formulations given Chapters 3 to 5 are self-contained and sufficient to address the corresponding problem, with each step it is aimed to achieve a more realistic design tool. Progressing through these three applications: a) the number of criteria decreases, b) the feasible design space gets smaller, either by increasing constraints or different sets of variables, c) the assumptions are removed for a more accurate evaluation. Chapter 6 summarizes this evolution and further analyzes the improvements through these steps.

CHAPTER 2
LITERATURE REVIEW

This chapter has two main focuses: background of the assembly synthesis, and the previous work on design for modularity. Though rigorous methods have not emerged until recently, both topics have been subject of numerous studies and industrial applications that set the framework of today's more complex practices.

2.1 Assembly Synthesis Overview

Assembly synthesis, as defined in this thesis, is an optimization problem, so it is a natural first step to look into the extensive literature on structural optimization in this overview (Section 2.1.1). Structural optimization community is observed to have neglected assembly issues to some extent, but another topic of recent research interest, assembly sequence generation, deserves attention with respect to the representations and optimization tools in use, although the addressed problem is different (Section 2.1.2). The review of methods that integrate DFA and DFM into the design process, especially in an automated way, is given in a separate section (Section 2.1.3). Finally, Section 2.1.4 analyzes the previous work on decomposition-based assembly synthesis, the study that this thesis is based on, as well as a conceptually very similar application that is limited to sheet-metals.

2.1.1 Previous work on structural optimization

The work on structural optimization started with proportioning the dimensions of the structures, and then advanced to varying geometry of structures (such as nodal

coordinates of skeletal structures) for an optimum design. Both rigorous and heuristic methods for topology optimization, i.e. the techniques that involve changing the entire design topology (for instance the number of holes in a structure) have developed quickly after early 1980's (Chirehdast *et al.* 1994). Modern structural topology design methods enable top-down synthesis of an optimal structure that fits within a specified design domain from the specification of loading and boundary conditions (Saitou and Yetis, 2000). For instance Bendsoe and Kikuchi (1988) used the homogenization method to solve for the optimal material distribution with a specified amount of material, for the stiffest topology. Chapman *et al. (*1994) describe genetic algorithm (GA) based structural topology optimization of finely discretized design domains. Shea and Cagan (1999) present the shape annealing method, which uses shape grammar rules with the simulated annealing algorithm to perform shape optimization of trusses. Starting with a random initial structure, topology exploration occurs by applying topology modification rules that transform configurations in the current design; metrics for design performance determine the search direction in the simulated annealing algorithm.

A specific area that needs special attention is homogenization-based topology optimization of multi-component structures. Contrary to most joint design applications that are concerned with the optimization of physical parameters associated with joints (such as stiffness), a limited number of studies suggest using the connection design space as a variable (Johanson *et al*., 1994; Chickermane and Gea, 1997; Jiang and Chirehdast, 1997; Li *et al*., 2001). These methods require overlapping extended design domains for each component and assume that location of each joint is given as a predefined input.

Even though the extensive work on topology optimization is inspiring, it can be observed that the essential problem of decomposing complex products into simpler components for assembly has not been addressed. Joint design in general appears to be overlooked in the structural optimization research and it is one of the objectives of this thesis to contribute to the literature in this field.

2.1.2 Relations to assembly sequence generation

While assembly synthesis has not gained the deserved attention, assembly sequence planning has been an active research field during the past decade. An assembly sequence involves an ordered set of assembly moves and assembly; a planner usually generates the feasible sequences as a first step and chooses the best one with respect to criteria such as ease or reliability of assembly, fixturing, gripping, least total assembly cost and assembly-line layout (Baldwin *et al.*, 1991).

In most of the solutions for automated assembly sequence generation, the geometric model of an assembly is created by describing the components and the spatial relationship among them. In recent years, features that combine geometric and functional information have been introduced in modeling and planning for manufacturing of complex assemblies (Eng *et al.*, 1999; van Holland and Bronsvoort, 2000).

Most algorithms described in the literature solve assembly sequencing problems by graph searching. Each joining of a component to another component or to a subassembly is called a *liaison*. The general approach is building a liaison diagram and generating all possible subassemblies by decomposing this graph (also called "cut-set" algorithms). Then the possible assembly sequences are evaluated based on the given constraints to determine the most suitable one (Mantripragada and Whitney, 1998; Whitney *et al.*, 1999). Such an exhaustive searching method requires substantial computational resources even for a simple structure. As a computational tool, genetic algorithms (GA) have proved successful in solving combinatorial and complex problems, such as finding a near-optimal assembly plan, with a reasonable execution time (Senin *et al.*, 2000; Lazzerini and Marcelloni, 2000).

Assembly sequence planners often emphasize that it is their objective to determine the sequence as early as possible in the design process to decrease the design lead-time and allow better solutions to emerge faster. However, most applications require

detailed information on the assembly, such as definition of the liaisons between all components in an assembly, which is rarely known at the conceptual design phase; actually it is the main problem assembly synthesis is meant to address. Still, the tested and proven tools like graph representation and optimization with GA can be used with slight modifications to carry out decomposition-based assembly synthesis before the sequence planning starts, as will be described in the following chapters of the thesis.

2.1.3 Integrating DFA and DFM into design optimization

While the tools from assembly sequence planning can be conveniently used to model the assembly and search for the optimal decomposition, determining the criteria for the evaluation of different decompositions is still an important problem.

In an attempt to increase the designers' awareness of manufacturing and assembly considerations, professional societies and some companies have published a number of DFM and DFA guidelines for a variety of processes. Typical examples are (van Vliet *et al.*, 1999):

- Maximize standardization (materials, design concepts, components, tools, fixtures, modular design)
- Select solutions that simplify manufacturing (shape, composition etc.)
- Choose solutions that enhance uniformity and parallelism
- Minimize the number of required resources

Popular DFM and DFA methods (see, for instance, Boothroyd *et al.* (1994)) suggest use of checklists and other specialized analysis tools in a systematic way to improve designs for manufacturability and assemblability. Besides the requirement of human experts, direct applications of these guidelines may be cumbersome for complex

products, so automated DFM and DFA systems, integrated with the design optimization process, have been subject to considerable research recently.

Adapting the classification by Gupta (1997), mainly three different scales of manufacturability can be identified in the literature:

1. *Binary measures* simply measure if a given set of design attributes is manufacturable or not, given the process capabilities.
2. *Qualitative and abstract quantitative measures* rank different designs based on their manufacturability by a certain production process. Most of the time these measures are subjective and hard to interpret, and in the case when the designer employs multiple manufacturability analysis tools, it becomes very difficult to compare and combine the ratings from the two systems to obtain an overall rating.
3. *Time and cost* give a measure of the effort required to manufacture a part given the specifications. Since all manufacturing operations have time and cost, ratings based on these measures can easily be combined. Also, because of the realistic evaluation they bring about, they can be conveniently used to aid management in making make-or-buy decisions.

In the context of this thesis, early applications make use of DFM guidelines that fall into first and second categories, which are much easier to implement when detailed information on the processes is not available. However as the method progresses it is aimed to carry out cost modeling and replace all qualitative measures with fabrication cost values for each decomposition.

As an inspiring application that combines DFM, DFA and design optimization of structural products, Yao *et al.*'s (1998) *aggregate weld product model* deserves special attention. Their method involves selecting a feasible welding method from a given set of

available processes, by checking various design and fabrication constraints such as weldability (considering the material type, orientation of the weldments and the metal thickness), joint strength, and the time and cost for joint preparation and set-up. Though no formal optimization method is defined and applicability of the system to complex products is not verified in this paper, many of the presented ideas come close to the goals of this thesis, especially with regard to guaranteeing the feasible combination of the joint type and the weld type, and using the combined criteria of structural strength and production cost.

2.1.4 Decomposition-based assembly synthesis

While it is not possible to find a thorough assembly synthesis method in the structural optimization and assembly related studies, several authors recognized the importance of the concept and suggested further studies. For instance Huang (1993) adopts a decomposition approach for assembly sequence generation, based on the assumption that a backward search that separates the final assembly into simpler parts will be more efficient than a forward search for the assembly sequence planning problem. His method is limited to simple applications without a formal optimization process, but he correctly concludes that some of the geometric knowledge and spatial relationship given *a priori* should be derived by applying geometric modeling and spatial reasoning functions for a more effective analysis (Huang, 1993). More recently De Fazio *et al.* (1999) point out that conventional DFA techniques ignore important combinatorial aspects of complex assemblies, most importantly subassembly partitioning, a similar problem to decomposition-based assembly synthesis.

Wang and Bourne's (1997) integrated system for the design and production of sheet metal parts is believed to be closest approach to assembly synthesis, at least in the conceptual sense. Their research describes an automated method to help designers

decompose bent sheet metal products into manufacturable parts. The focus is on the cutting, bending, and assembly processes of the production of bent sheet metal products. The first step in this method is the manufacturability analysis of the end product and the decision to decompose if the product is not manufacturable in one piece. The process is recursive until all decomposed parts are manufacturable and evaluates the alternative decompositions with respect to manufacturing cost. The resulting decomposition is also to be checked by process planners to ensure its manufacturability; so the outputs are the geometric representation and the production plan of each resulting part (Wang and Bourne, 1997). This study is mainly geometry based optimization and may be considered somewhat limited in scope due to lack of structural strength criteria, but the method still effectively demonstrates that efficient exploration of decompositions at the conceptual design stage expedites the entire design process.

The project that this thesis is based on was originated in the Discrete Design Optimization Laboratory at the University of Michigan in 1999-2000. The objective was to develop a systematic decomposition process for *assembly synthesis*: "decision of which components are better to assemble together to achieve a certain end product" (Saitou and Yetis, 2000; Yetis, 2000; Yetis and Saitou, 2002). Formulated as an optimization problem, the method has two major steps:

1. A two-dimensional bitmap image of a structure (may be obtained via structural topology optimization) is transformed to a product topology graph through application of image processing algorithms.
2. The product topology graph is decomposed into subgraphs by using GA, which results in a decomposition of the actual product with optimal joint attributes.

During the product topology graph generation, members of the structure are mapped to nodes and the intersections are mapped to multiple edges since they can be

joining more than two members. The search for optimal decomposition can then be posed as a graph-partitioning problem, a discrete optimization task, as the problem is defined over a set of discontinuous states (edges to be cut by a partition). In summary, the problem is: given the topology graph of the structure, obtain the partition representing the optimal decomposition and the corresponding joint attributes, subject to a cost function evaluating the decomposition quality.

In this problem, the only available joining method is determined to be a spot weld and the design variable for each such joint is the weld angle to be chosen from discrete set of possible values. The objective function evaluates each decomposition with respect to structural strength and assemblability. The reduction in structural strength is defined as the sum of tensile stresses on all welds, which should be minimized. For assemblability, the similarity of weld angles and the number of welds in the decomposition are taken into account. Along the line of DFM and DFA rules, it is assumed that lower number of welds is preferred to simplify the assembly process and minimize the required resources. Similar weld angles are favored to end up with higher uniformity and parallelism in the design.

As mentioned in Chapter 1 to be one of the important goals of this thesis, deficiencies of the early versions of assembly synthesis will be addressed. The method will make use of more realistic joint models for increased applicability, a manufacturability criterion that is missing in the original formulation will be introduced and the optimization will be improved by modifying the genome representation as well as the operators. Further details will be given in the following chapters.

2.2 Modularity and Design of Product Platforms/Families

Systematic methods to implement the modularity concept have started to emerge only recently, but interest is getting widespread, in several engineering disciplines, as

well as the management science community. Section 2.2.1 first aims to clarify the terminology used for the modular design process in the literature. Early studies and related methodologies such as Group Technology (GT) are also reviewed. There are two main approaches for systematic modularity analysis and both are examined comprehensively in Section 2.2.2. Finally, several researchers incorporate cost estimation to quantify the benefit of modularity, parallel to the objectives of this thesis, and these studies are summarized in Section 2.2.3.

2.2.1 Terminology and early studies on modularity

In the literature, during the *design of a product family,* a shared subsystem with common components and interfaces is often referred to as a *product platform*. The term 'product platform' usually implies that a number of *variant designs* can be developed quickly based on this common architecture. This can be true for some special products and relatively simpler applications, but in the context of this thesis, main focus is on complex structural products; it is presumed that the different designs are created based on different needs, and component commonality is considered as an optional strategy, used only if beneficial. It is believed that in such a design process, shared components should be termed as *modules* instead of platforms, and the method is consistently called *design for modularity* throughout this thesis. There are no fundamental conceptual and methodological differences between design for modularity and design of product platforms, so this terminology is not expected to lead to any ambiguity.

A module can be defined as a component produced in high volumes and used in a family of products, as often suggested in the literature; or can be considered parts that are similar but not exactly the same, which can share some or all processing equipment, a definition adopted in the industry. This study aims to develop a method which can be conveniently used for both approaches.

Though widespread applications are quite recent, the idea behind modularity is not new; benefits of component sharing have been recognized for a long time. The concept of Group Technology (GT), as a management philosophy, is the application of knowledge about groups, more specifically, it is the notion of recognizing and exploiting similarities in a production environment. GT involves standardization, performing like activities together, as well as storing and retrieving information about recurring problems. It is believed that the application of the GT principles in design tasks paved the way for the effective application of modularity concepts. It is reported that early users of GT in the design engineering community were far less in number compared to process planners, but this has obviously changed as the organizational and personal barriers that separate design and manufacturing functions are eliminated. Users of GT in design retrieve previously designed parts from a database using specialized GT coding and classification schemes, to check if an existing design can be used in a new product instead of creating a new part. Even though systematic and rigorous methods to aid the designers in this task were absent, it was known that 40% of new design requests could be satisfied by existing parts, so potential benefits of modularity were well understood (Hyer and Wemmerloev, 1989).

One early example of modularized products is the Nippondenso case; the Japanese company was able to offer 288 unique product variants to their customers. Another example is the Black & Decker case in the early 70's when their power tools were modularized (Stake, 1999). In the 1990's, companies such as Toyota and Sony successfully applied the principles of modularity to reduce their design complexity (Kota *et al.*, 2000). More recently, Volkswagen is reported take advantage of component commonality by sharing between its four major brands, namely VW, Audi, Skoda, and Seat. Ford Motor Company has similar ambitions within its new Generic Architecture Process program: common suspension systems and drivetrains are planned to be used in several car models. Other carmakers are expected to be using similar approaches to

design the essential features of a family of cars at the same time, instead of one model at a time (Dahmus *et al.*, 2001).

2.2.2 Single-stage vs. two-stage methods for modular design

A number of researchers propose numerous metrics, design charts and commonality indices (Ishii, 1998; Newcomb *et al.*, 1998; Kota *et al.*, 2000) to carry out an interactive design for modularity, but these formulations can hardly be used for complex products, which require the integration of modularity into an optimization formulation to explore all possible component sharing alternatives.

Similar to the classification in (Simpson and D'Souza, 2002), two different approaches for the optimal solution of the design for modularity problem can be identified. In *two-stage approaches,* first stage of the optimization is devoted to optimally selecting the components to be held common, followed by instantiation of the individual products during the second stage. Alternatively, eliminating the partition between two different optimization processes, *single-stage approaches* optimize the components to be shared and resulting family of products simultaneously, bringing in a serious computational burden due to higher dimensionality. Despite the additional challenges, a single-stage optimization approach is the preferred method in the design for modularity; note that two-stage approaches require *a priori* decisions on sharing of design variables at a time when the designer does not really know about the relative impact of the variables on each product's performance (Simpson and D'Souza, 2002).

Zugasti *et al*. (2001) started with several predefined module alternatives and design variants in their two-stage formulation. The optimal product family can then be identified based on decision analysis and real options; modeling the risks and delayed decision benefits present during product development. Another example of two-stage optimization is reported by Nelson *et al*.(2001), where a multi-criteria optimization

problem is formulated with a solution that quantifies the performance degradation of product variants by component sharing. For each selection of modules to share, the performance trade-offs between two product variants are represented as a Pareto curve. Several such Pareto curves are shown to illustrate the effect of different module selections on the performances of the two product variants. While the optimal module design should be on one of these Pareto curves, exactly which one is the best is a question of performance as well as of other business issues.

Simpson *et al*. (2001) focused on a slightly different two-stage application, introducing the Product Platform Concept Exploration Method to design scalable modules and the resulting product variants. The goal is to design a product that can be vertically leveraged for different market niches. Some parameter values in the modules are shared across given product variants, while other parameters can take different values within each product variant. A group of individually optimized products are compared with the product variants with shared modules and it is reported that component sharing is achieved without a considerable loss in performance.

Examples of single-stage approaches have started to appear in the literature only recently. One of the earlier published studies on single-stage design for modularity is by Fujita and Yoshida (2001). They present a hybrid optimization method combining a genetic algorithm, a branch-and-bound method and a nonlinear programming algorithm. In their three-level technique, they first optimize the module selection and similarity among different products using genetic algorithm, then optimize the directions of similarity on scale-based variety using branch-and-bound, and finally optimize the module attribute with sequential quadratic programming (SQP).

In Simpson and D'Souza's work (2002), a genetic algorithm with binary and continuous variables is employed for product family design. In this formulation when a binary variable takes a value of 1, then the corresponding continuous design variable is made common among all of the products in the family; a value of 0 makes that design

variable unique among the products. Fellini e*t al.* (2002) present a similar approach using binary decision variables for alternative sharing strategies, but instead of having objective function terms for performance optimization, they introduce constraints to make sure the performance loss does not exceed a user-specified tolerance. To avoid the combinatorial nature of the problem, they propose a continuously differentiable approximation to the sharing function, allowing the use of gradient-based algorithms.

Extending the work of Simpson *et al.* (2001), Messac *et al.* (2002) introduce a product family penalty function based on physical programming for the simultaneous determination of common variables and scaling variables that generate the product family. They propose that design variables with minimal effect on the design objectives can be shared and set at a constant value for all products; so essentially this method optimizes each performance design objective while minimizing the variations of all the design parameters considered to be scaling variables.

All these studies present effective methods to solve the class of problems they are concerned with, however it can be observed that there are two major recurring assumptions that this thesis aims to discuss and ultimately eliminate:

1. *Components/variables with sharing potential are assumed to be known*: Note that in two-stage methods the shared components/variables have to be chosen before starting the optimization process (second stage), and even in single-stage approaches at least a set of components/variables that is recommended for sharing is given *a priori*. Incorporating a sharing potential check, essentially a geometric similarity comparison, into the optimization formulation, all possibilities of sharing can be examined during the process and no input from the user is required for this purpose.

2. *Benefit of component sharing is taken for granted*: The basic premise in design for modularity is that the component sharing would result in less design effort and fewer

production varieties with higher volumes, hence reducing overall production cost. However, component sharing has a tendency to result in overdesign of low-end products and more importantly, underdesign of high-end products in a product family. This effect, therefore, has to be outweighed by the economical gain of component sharing to justify a decision on component sharing. As to be described in next section and Chapter 5, incorporating a production cost model, modularity will be an outcome rather than a premise, so this assumption will not be in use any more.

2.3 Cost Estimation and Component Sharing

Since component sharing often results in compromise in the performance of individual products, it is essential to quantify its effect on the overall production cost, in order to effectively assess the trade-offs between cost savings due to sharing and performance compromises.

Kim and Chhajed (2000) developed an economic model that considers a market consisting of a high segment and a low segment. Greater commonality decreases production cost but makes the products more indistinguishable from one another, which makes the product more desirable for the low segment but less desirable for the high segment. Although the quality provided through the common design will yield the same utility, they report that there is a valuation change due to the product similarity, which affects the perceived quality of products. On the supply side, cost saving will occur if a common modular design is used for the design of multiple products. The article analyzes several sharing strategies using the cost model but does not suggest a rigorous solution for the optimization problem at hand.

Meyer *et al.* (1997) propose measurement methods of R&D performance during platform design. One measure is called *platform effectiveness*; the degree to which the products based on a product platform produce revenue for the firm relative to the cost of

developing those products. Mathematically, platform effectiveness considers R&D returns as accumulated profits divided by development costs, either at the individual product level, or for groups of products within distinct platform versions. They present a real life application, but the method is used essentially for analysis of different sharing alternatives, rather than a tool during design.

Fisher *et al.*'s work (1999) appears to be very close to our application in this paper. They present an analytic model of component sharing and show through empirical testing that this model explains much of the variation in sharing practice for automotive braking systems. The model takes as inputs a set of cars for which brakes must be designed and a set of possible design alternatives, and determines which versions of each component should be built and which cars should use each component version to minimize cost. The cost functions model fixed and variable costs, and nonlinear production economies of scale. This discrete optimization formulation is then converted to a shortest path problem. Similarly, Fujita and Yoshida (2001) use a monotonic cost model for the assessment of benefits of commonality. The model consists of design and development cost (proportional to weight of each module), facility cost (proportional to a representative attribute) and production cost (composed of material cost and processing cost). A learning effect is incorporated by reducing the production cost in accordance with increasing number of production units due to commonality.

Conceptually, all these methods address the problem accurately, however, in terms of the applications there is room for improvement. The researchers that incorporate cost modeling tend to use simplistic optimization procedures that consist of comparing different commonality scenarios, or apply their formulations on very simple products only.

The specific cost evaluation approach to be used in this study on automobile space frame structures in Chapter 5 is based on the technical cost modeling method

developed at MIT Materials Systems Laboratory (Clark, 1998, Kelkar *et al.*, 2001; Constantine, 2001). Kelkar *et al.* (2001) report that the manufacture of the body-in-white is comprised of two costs: fabricating the parts and assembling the parts, with inputs of design specifications, material parameters, and production parameters. Inputs are transformed into estimates of fixed and variable costs for each manufacturing step. Variable costs include energy, materials, and direct labor; fixed costs cover capital equipment required for the manufacturing process, building expenses, maintenance etc. They present the change of the average cost of each part with respect to production volume (Figure 2.1), which indicates the main motivation for sharing modules in a family of products: it is possible to go down the curve by increasing the total production of the components and achieving considerable cost reduction. This cost model is able to offer the desired accuracy in assembly synthesis and the tendency given in the curves will be conveniently used to quantify the benefit of modularity due to increased volume of shared components.

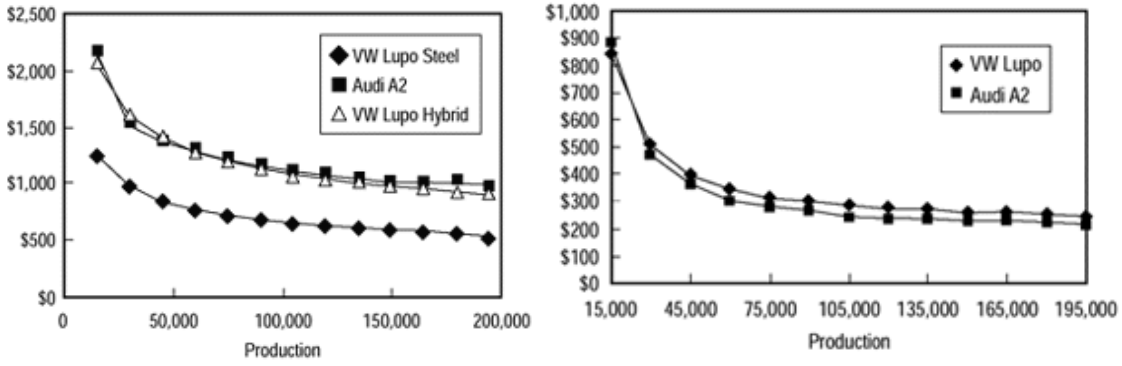


Figure 2.1. Fabrication and assembly costs for several automobile body structures, from (Kelkar *et al.*, 2001).

CHAPTER 3

ASSEMBLY SYNTHESIS FOR CONTINUUM-BASED PRODUCTS

3.1 Introduction

In the adaptation of the assembly synthesis method to multiple continuum-based products, image of a structure obtained via structural topology optimization is decomposed automatically into an assembly consisting of multiple structural members with simpler geometries. Note that as topology optimization is an entirely separate process from the assembly synthesis, any image, not necessarily topologically optimal, can be the input for this system.

The formulation given in this chapter is very similar to the early work on assembly synthesis (Yetis and Saitou, 2000; Yetis and Saitou, 2002), the only essential difference is the additional objective function term for modularity evaluation. There are two main steps in this process:

- A two-dimensional bitmap image of a structure obtained via structural topology optimization is transformed to a product topology graph through application of image processing algorithms.
- The product topology graph is decomposed into subgraphs by using a genetic algorithm which results in an optimal decomposition of the product with respect to structural strength, assemblability and modularity criteria, choosing one of the available weld angles for each joint or the option of ‘no weld’.

The following flowchart illustrates the main steps of the assembly synthesis method for continuum-based products:

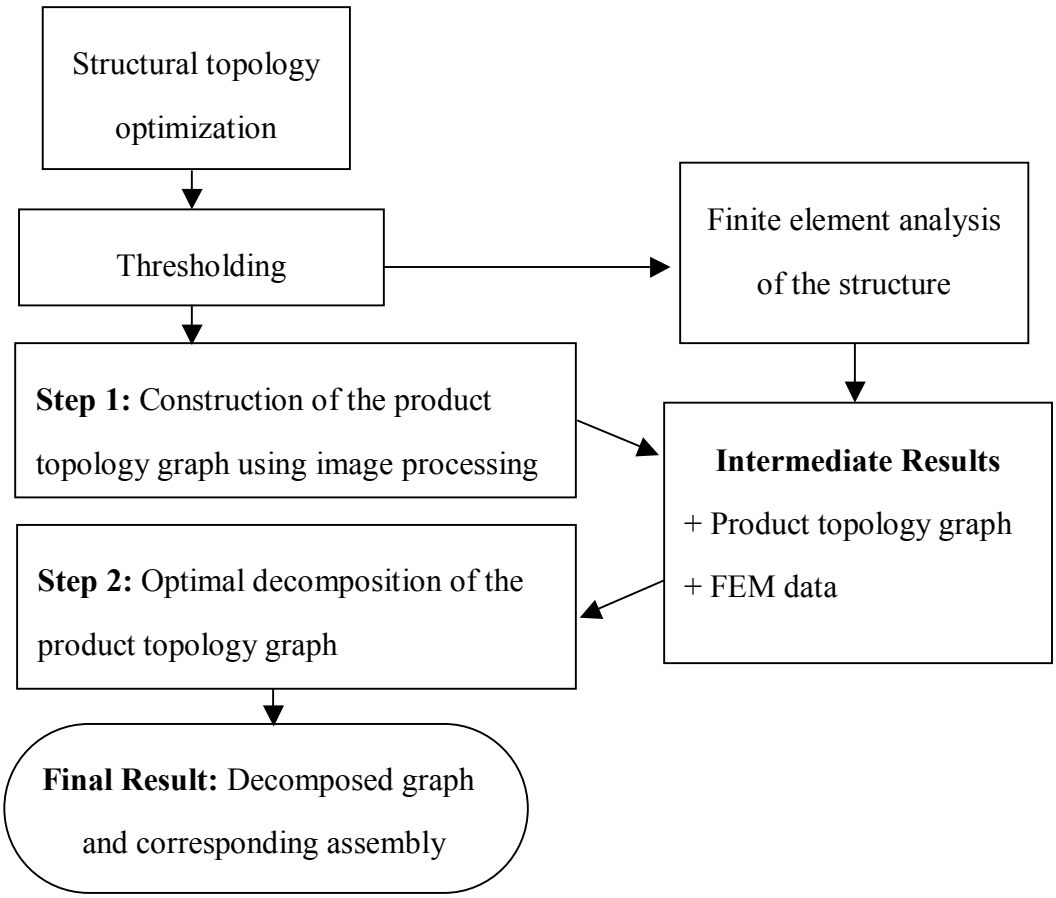


Figure 3.1. Flowchart of the decomposition-based assembly synthesis method.

Details of the structural topology optimization procedure will be omitted in this study. Image processing algorithms will be also very briefly discussed, details can be found in (Yetis, 2000). The major operations are:

Dilation

It fattens the image by filling small, isolated holes and expanding the image boundary.

Skeletonization

It has an opposite effect to dilation. It thins the image by expanding small, isolated holes and shrinking the image boundary.

Hough transform

This procedure detects lines in a bitmap image by mapping the image in the *x*-*y* space to a parameter space (the θ-ρ space) using the normal representation of a line in *x*-*y* space. Since a pixel (x_i, y_i) in *x*-*y* space corresponds to a sinusoidal curve $x_i\cos\theta + y_i\cos\theta = \rho$ in the θ-ρ space, collinear pixels in the *x*-*y* space have the intersecting sinusoidal lines in the θ-ρ space. Conversely, an intersection point (θ_n , ρ_n) in the θ - ρ space corresponds to a line in the *x*-*y* space. Therefore, all lines passing through arbitrary pairs of pixels in the image are found by checking the intersection points in the θ - ρ space.

A simple example to illustrate both interactive and automated image processing operations is given in Figure 3.2.

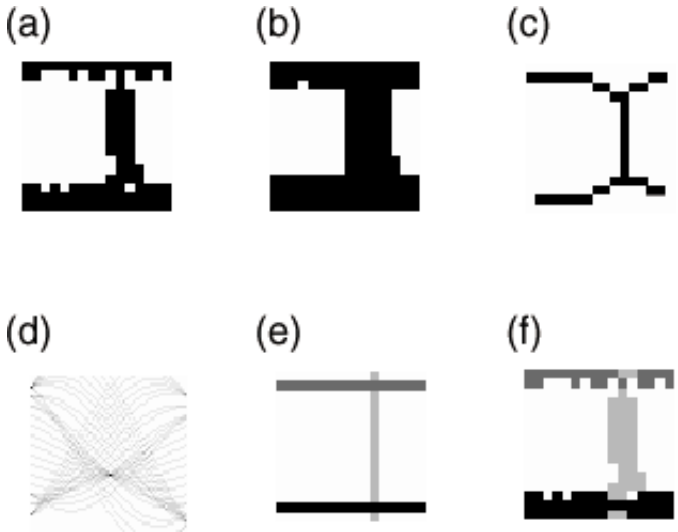


Fig. 3.2. An example of product topology extraction: (a) original image, (b) dilation, (c) skeletonization, (d) initial Hough transform (shown in θ-ρ space), (e) primary line extraction, and (f) topological segmentation (Yetis, 2000).

Once the lines are detected and topological segmentation according to this extraction is performed, the graph corresponding to this configuration can be developed. During the product topology graph generation, members of the structure are mapped to nodes and the intersections are mapped to multiple edges since they can be joining more than two members.

Note that at this point removing edges in the topology graph will correspond to dividing the structure itself into simpler components. The search for optimal decomposition can then be posed as a graph-partitioning problem, a discrete optimization task, as the problem is defined over a set of discontinuous states (edges to be cut by a partition).

In the current problem it is decided that joining method at every joint is assigned as a spot weld and the only joint feature considered is the weld angle which is chosen from discrete set of possible values. Since spot weld joints are much weaker against tensile loads than against shear loads (Hahn *et al*. 1997), to evaluate the decomposition according to the structural strength criteria, the normal stress at the joints and the area on which the normal stress acts are calculated. The evaluation is based on the difference between the angle at which the normal stress is minimum (θ_{ideal}) and the chosen mating angle; note that deviation from the ideal angle means higher normal stress. So in this formulation maximum structural strength is achieved when there is no weld (no decomposition case) or when none of the welds are under tensile stress. Consequently the reduction in structural strength is defined as the sum of normal stresses on all welds, which should be minimized.

When assemblability is considered, the similarity of weld angles and the number of welds in the decomposition are taken into account. Along the line of DFM and DFA rules, it is assumed that lower number of welds is preferred to simplify the assembly process and minimize the required resources. Similar weld angles are favored to end up with higher uniformity and parallelism in the design.

The modularity criteria proposed in this work is implemented by analyzing two structures at a time, and assessing the similarity of the disconnected components to point at a probable part commonality. A term is added to the objective function to favor the decompositions that result at: a) components with similar stress states, represented by the joint angles, b) components that are geometrically similar to each other, by considering the lengths and thicknesses of their corresponding members, or by using an equivalent measure of shape similarity. Also, before evaluating the cost function component related to modularity, it is certified that the subgraphs of the components to be shared are *isomorphic*; note that this is a necessary but not sufficient condition for two structures to have the same configuration.

Thus the final objective function attempts to find a solution that results in two decompositions with maximum structural strength, maximum assemblability, and one or more components that can be shared by the both designs. To be used as an illustrative case throughout this section, consider the two simple structural design problems given in Figure 3.3: note that the only the difference between (a) and (b) is the application point of the concentrated force P = 1000 N.

(a) Design problem 1 (b) Design problem 2

Figure 3.3. The design problems to be addressed simultaneously.

The structural topology optimization process[i] is carried out for the loadings given in Figure 3.3, aiming to achieve a structure with maximum stiffness using 40% of the volume in the design domain. The optimization method is based on 'power-law approach', in which relative density is the only design variable, and upper and lower bounds of the relative densities, the volume fraction and the equilibrium conditions are the constraints (Sigmund, 2001). After thresholding the results, the topologies presented in Figure 3.4 are obtained. So the problem becomes finding two optimal decompositions for Figures 3.4 (a) and (b) so that maximum structural strength for both structures are maintained, and at the same time some components are shared by the products.

(a) Topology for problem 1 (b) Topology for problem 2

Figure 3.4. Optimum topologies for the design problems.

Note that the formulations and examples given in the remaining sections of this chapter are also presented in (Cetin and Saitou, 2001) and (Cetin and Saitou (a)).

[i] The web-based topology optimization software at the Technical University of Denmark (http://www.topopt.dtu.dk) is used.

3.2 Mathematical Model

3.2.1 Definition of the design variables

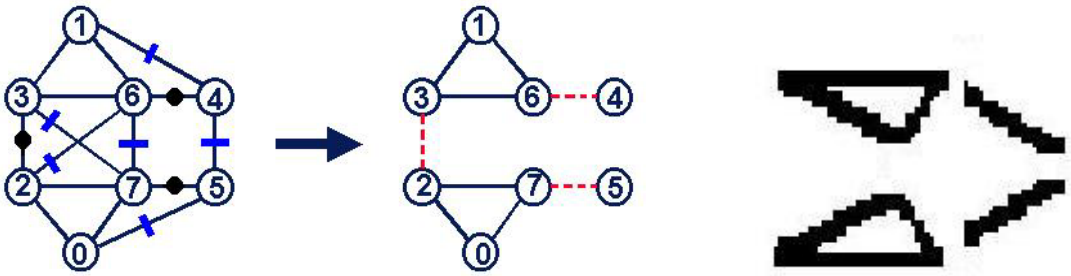
Let the members of the structure be mapped to the nodes of the product topology graph and the intersections be mapped to the edges[ii]. This mapping is done automatically using the image processing tools described in Section 3.1. The graph representation for the optimum topology of the first design problem is given in Figure 3.5 (b) as an example. So the whole structure can be represented as $G=(V, E)$ with a node set V and an edge set E. The problem of optimal decomposition becomes one of finding a partition, *i.e.,* the design variable P, of the node set V such that the objective function, $c(P)$, is maximized.

The optimal partitioning of G can be represented mathematically by a vector $\boldsymbol{x} = (x_i)$ where x_i is a binary variable representing the presence of edge e_i in the decomposition defined by the partitioning P. It is obvious that $i=1,\ldots,|E|$ since there are $|E|$ edges in the topology graph. Another vector $\boldsymbol{y} = (y_i)$ is defined to store the mating features for each edge e_i; note that domain of $\boldsymbol{y}$ depends on the model of the joint represented by the edge. A set, J, of joint features is therefore has to be defined. Based on assumptions in the earlier work, J is the set of possible mating angles at the welded joints.

As an illustration of the concept, consider Figure 3.5 (c), where a certain partitioning is depicted; all the marked edges are cut in the graph, and the ones with a circular spot are to be welded (also shown as dashed lines in the second part of Figure 3.5 (c)). Figure 3.5 (d) presents the corresponding decomposition.

[ii] LEDA library developed at the Max-Planck Institute of Computer Science (http://www.mpi-sb.mpg.de/LEDA/) is used for the graph algorithms.

(a) Simplified topology (b) Corresponding graph



(c) A sample partitioning (d) Corresponding decomposition for (c)

Figure 3.5. Graph representation for problem 1.

3.2.2 Definition of the constraints

The constraint on the vector $\boldsymbol{x}$, which represents the presence of edges, is the following:

$$\text{COMPONENTS}(\text{GRAPH}(\boldsymbol{x})) = k \tag{3.1}$$

where

- GRAPH($\boldsymbol{x}$) returns the graph after the edges with $x_i = 0$ in vector $\boldsymbol{x}$, have been removed from the original topology graph,

- COMPONENTS(G) returns the number of disconnected components in graph G,
- k denotes the desired number of components specified by the user.

The constraint on vector $\boldsymbol{y}$ is as follows:

$$y_i \in J \tag{3.2}$$

where J is the set of mating angles at which spot welds can be applied at the joints. One element of set J represents the case for no weld at the corresponding joint.

Another constraint is imposed on the combination of the vectors $\boldsymbol{x}$ and $\boldsymbol{y}$ in the following way:

$$\text{IS\_CONNECTED}(\text{COMBINED\_GRAPH}(\boldsymbol{x}, \boldsymbol{y})) = 1 \tag{3.3}$$

where

IS_CONNECTED(G) is a function which returns 1 if the graph G is connected and returns 0 otherwise.

COMBINED_GRAPH($\boldsymbol{x}$,$\boldsymbol{y}$) is a function that returns a graph which consists of the nodes of the original graph and the edges in vectors $\boldsymbol{x}$, $\boldsymbol{y}$. This constraint ensures that the combination of the decomposition given by vector $\boldsymbol{x}$ and the mating angles given by vector $\boldsymbol{y}$ constitutes a structure which has the same connectivity as the original disconnected structure.

3.2.3 Definition of the objective function

Objective function will evaluate each decomposition according to the following criteria:

- Reduction of structural strength due to introduction of joints
- Assemblability of the decomposed structures
- The maximum modularity of the structures

To evaluate the decomposition according to the structural strength criteria, the normal stress at the joints and the area on which the normal stress acts are calculated. The evaluation is based on the difference between the angle at which the normal stress is minimum, θ_i^{ideal}, and the chosen welding angle given by vector $\boldsymbol{y}$, as deviation from the ideal angle means higher normal stress. The stress at the chosen angle multiplied by the weld area provides a measure of force acting on the weld which is also used in evaluating the decrease in strength. A weld with larger area introduces a higher amount of decrease in strength than a weld with smaller area.

While assessing the decomposition with respect to the assemblability criteria, the similarity of weld angles and the number of welds in the decomposition are taken into account. Obviously, lower number of welds and similar weld angles result in higher assemblability.

These criteria result in the following objective function component for structural considerations:

$$f_S(\boldsymbol{x},\boldsymbol{y}) = w_1 \sum_{i=1}^{N_{welds}} \left(\theta_i - \theta_i^{ideal}\right)^2 + w_2 \sum_{i=1}^{N_{welds}} \left(\sigma_i(\theta_i) A_i(\theta_i)\right) + w_3 \sum_{i=1}^{N_{welds}} \sum_{j=i+1}^{N_{welds}} \left(\theta_i - \theta_j\right)^2 + w_4 N_{welds} \tag{3.4}$$

The variables are defined as follows:

- $\boldsymbol{x} = (x_i)$: x_i is a binary variable representing the presence of edge e_i

- $\boldsymbol{y} = (y_i)$: y_i is discrete variable representing the choice of weld angle at joint i
- $w_{j:}$ weight of j-th criteria in the objective function
- N_{welds}: total number of welds in the decomposed structure
- θ_i: weld angle with respect to vertical direction at joint i
- θ_i^{ideal}: angle of minimum normal stress at joint i
- $\sigma_i(\theta_i)$: normal stress at joint i at angle θ_i
- $A_i(\theta_i)$: weld area at joint i (function of θ_i)

As the second part of the objective function, the cost function for modularity is incorporated to evaluate two attributes of the components to be shared between the structures:

- Similarity in stresses that the components are subject to: this condition is simply implemented by maintaining that joint angles of the components should be close to each other,
- Similarity in shapes of the components in a given (user-specified) tolerance: this attribute is checked by comparing the components with respect to their areas.

Note that this procedure requires that all components that come out of the decomposition process of one structure be compared with the components in the second design problem. However, probably only a few of the components at each iteration will have the same number of members assembled in a similar manner. Thus, before evaluating how similar two components are, it is convenient to test if the corresponding subgraphs are *isomorphic*: the modularity cost function should return a large number if no components are found to be *isomorphic*, and if this check is passed, then the similarity measure can be applied. Considering the computational overhead of this check, a simple approximation, actually a necessary but not sufficient condition is utilized in the

software: it is required that the components have an equal number of nodes and edges to be shared. A fast *graph isomorphism* check algorithm will be employed for more complex design problems in the future work.

Thus the modularity component of the objective function is implemented as follows:

$f_m(x_1, y_1, x_2, y_2)$ (3.5)

1. cost = 0, module = 0
2. **for** each pair of subgraphs (g_1^k, g_2^l)
3. **if** IS_ISOMORPHIC(g_1^k, g_2^l) = TRUE
4. module = module + 1
5. cost = cost + $w_5 \sum_{i=1}^{N^c_{welds}} ((\theta_1)_i - (\theta_2)_i)^2 + w_6\, h(g_1^k, g_2^l)$
6. **if** module = 0
7. **return** a large number
8. **else**
9. **return** cost

where

- g_1^k and g_2^l are two subgraphs representing the component in structures 1 and 2 decomposed as specified in $\boldsymbol{x}_1$, $\boldsymbol{y}_1$ and $\boldsymbol{x}_2$, $\boldsymbol{y}_2$, respectively.
- Superscripts k and l are the indices of subgraphs in each structure.
- w_5 and w_6 are the weights for the corresponding criteria,
- $(\theta_1)_i$ and $(\theta_2)_i$ are the weld angles at joint i of each component,
- N^c_{welds} is the number of welds in the shared components,
- IS_ISOMORPHIC(g_1^k, g_2^l) is a function that returns TRUE if subgraphs g_1^k, g_2^l are *isomorphic*, FALSE otherwise. For the time being the function only

checks if the two subgraphs have the same number of nodes and edges. Therefore the size of the combinatorial problem that results from checking each pair of subgraphs from the structures decreases considerably.

- $h(g_1^k, g_2^l)$ is a function that returns a measure of geometric similarity between the components. This measure can be realized by the calculation of first moments of component areas with respect to the centroids, or the origin, if the locations of the components in the configuration are to be incorporated.

Note that before $f_m(\boldsymbol{x}_1, \boldsymbol{y}_1, \boldsymbol{x}_2, \boldsymbol{y}_2)$ returns a cost at an iteration, all components, i.e. all subgraphs are examined, and only if none of them are *isomorphic* a large number is returned to introduce a penalty for lack of part commonality. If more than one component in each structure match with others, the similarity measures can be easily modified to favor the sharing of several components among the products.

3.2.4 Formulation of the optimization problem

The constraints and objective function combine to give the following optimization problem:

$$\text{minimize } f(\boldsymbol{x}_1 \boldsymbol{y}_1, \boldsymbol{x}_2, \boldsymbol{y}_2) = f_s(\boldsymbol{x}_1, \boldsymbol{y}_1) + f_s(\boldsymbol{x}_2, \boldsymbol{y}_2) + f_m(\boldsymbol{x}_1, \boldsymbol{y}_1, \boldsymbol{x}_2, \boldsymbol{y}_2) \tag{3.6}$$

subject to

$$\boldsymbol{x_1} \in \{0, 1\}^{|E_1|}$$

$$\boldsymbol{x_2} \in \{0, 1\}^{|E_2|}$$

$$\boldsymbol{y_1} \in \boldsymbol{J}^{|E_1|}$$

$$\boldsymbol{y_2} \in \boldsymbol{J}^{|E_2|}$$

$$\text{COMPONENTS}(\text{GRAPH}(\boldsymbol{x_1})) = k_1$$

$$\text{COMPONENTS}(\text{GRAPH}(\boldsymbol{x_2})) = k_2$$

IS_CONNECTED(COMBINED_GRAPH(x_1, y_1)) = 1

IS_CONNECTED(COMBINED_GRAPH(x_2, y_2)) = 1

3.3 Optimization Method

3.3.1 Genetic algorithm formulation

The exact solution of the graph partitioning problem, even with a linear objective function, requires exponential computation (Garey and Johnson, 1979). Noting the computational overhead and high non-linearity of the cost function, a genetic algorithm (GA), is conveniently used in this project. GAs are regarded as a compromise between random and informed search methods, and they have proved very efficient in the solution of discrete optimization problems.

The decomposition problem is to be solved by using a steady-state GA (Davis, 1991). Instead of replacing all parents by their children as in conventional (generational) GA, this approach involves keeping a specified percentage of the population and renewing the rest with the newly formed chromosomes. The basic flow of the algorithm is given below. Note that the fitness is assumed to be *minimized*:

Steady-state Genetic Algorithm

1. Randomly create a population P of n chromosomes (an encoded representation of design variables) and evaluate their fitness values and store the chromosome with the *minimum* fitness value. Also create an empty subpopulation Q.
2. Select two chromosomes c_i and c_j in P with probability proportional to ($f_{max} - f_i$) and ($f_{max} - f_j$), respectively, where f_i *and* f_j *are* the fitness value of chromosome c_i *and* c_j, and f_{max} is the maximum fitness in P at the current generation.
3. Crossover c_i and c_j to generate two new chromosomes c_i' and c_j'.

4. Mutate c_i' and c_j' with a certain low probability.
5. Evaluate the fitness values of c_i' and c_j' and add them in *Q*. If *Q* contains less than m new chromosomes, go to 2.
6. Replace m chromosomes in P with the ones in *Q* and empty *Q*. Update the best chromosome and increment the generation counter. If the generation counter has reached a pre-specified number, terminate the process and return the best chromosome. Otherwise go to 2.

Empirical advantages of steady-state GA are that it prevents premature convergence of population and reaches an optimal solution with fewer number of fitness evaluations (Davis, 1991). These improvements can be attributed to the fact that child chromosomes can mate with their parents in subsequent steps of the steady-state procedure, leading to better solutions faster. The resulting children are also often checked against the remaining parents to avoid duplication and possible early domination.

Each solution is encoded in a chromosome in the following way: The chromosome is of length $2|E|$ where $|E|$ is the number of the edges in the graph. First $|E|$ genes carry binary information about which edges of the topology graph are kept and which are removed to produce a decomposition (Figure 3.6). If the i^{th} element of the chromosome is 0, it means that this edge has been cut in this particular decomposition represented by this chromosome.

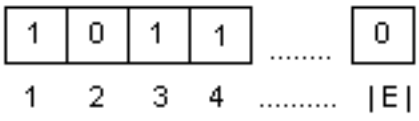


Figure 3.6. First half of chromosome with binary information.

The second half of the chromosome carries the information about which discrete choice of possible mating angles is chosen for a given joint (Figure 3.7). The $(|E|+i)^{th}$ element carries the choice of mating angle for the i^{th} joint(edge in the graph).

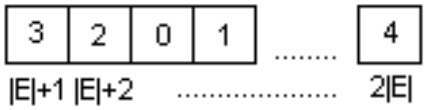


Figure 3.7. Second half of chromosome with mating angle information.

Since the procedure introduced in this project requires the simultaneous evaluation of two structures, apparently the chromosomes given in Figures 3.6 and 3.7 cannot be used on their own. A simple way of examining two chromosomes, i.e. two partitioning problems at once is combining the chromosomes and treating them properly by customized crossover and mutation operations. Then the length of the chromosome becomes $2|E_1| + 2|E_2|$, where E_1 and E_2 represent the number of edges in each structure's topology graph. Since the customization of the operations and representations given in this section for this new application only involves the repetition of the tasks for both 1st and 2nd structures, and the implementation consists of solely changing the indices to point to the correct gene, details are avoided in this paper.

For this study, the possible mating angles have been chosen as –45, 0, 45, 90 degrees from the vertical and map to gene values of 1, 2, 3, 4, respectively, as given in Figure 3.8. A gene value of zero means no weld at that intersection.

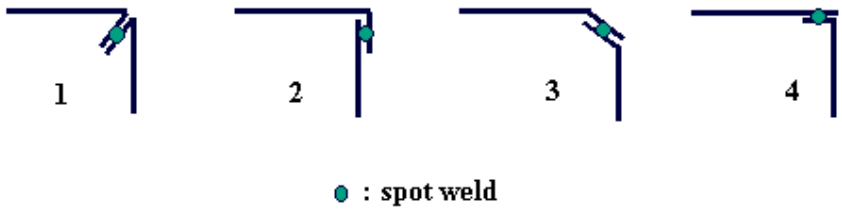


Figure 3.8. Possible mating angles at the joints.

Since chromosomes representing the decompositions carry two different kinds of information (x_i is binary and $y_i \in J$) the cross-over and the mutation operators have been customized. The crossover operator treats the first and second halves of the chromosome simultaneously since the information in the second half complements the information in the second half and only combinations of corresponding genes in the first and second halves represent a good or bad solution. Therefore application of crossover at the same point in both halves preserves the good or bad nature of the chromosome. Practically the custom crossover operator is a multi-point crossover operator (Figure 3.9).

As genetic algorithms do not handle constraints directly, the constraints in the mathematical problem formulation have to be translated into penalty terms. Therefore, the fitness function will consist of two main terms; the objective function value $f(\boldsymbol{x}_1, \boldsymbol{y}_1, \boldsymbol{x}_2, \boldsymbol{y}_2)$ of the decomposition and the penalty term which imposes the constraints of the mathematical model:

$$\text{Fitness} = f(\boldsymbol{x}_1, \boldsymbol{y}_1, \boldsymbol{x}_2, \boldsymbol{y}_2) + \text{Penalty terms} \tag{3.7}$$

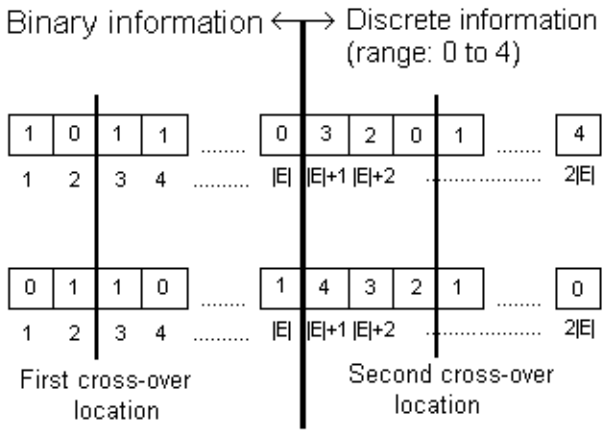


Figure 3.9. Crossover of two chromosomes.

The constraint on vectors $\boldsymbol{x}$ and $\boldsymbol{y}$ are imposed simply by the chromosome representation of the problem, *i.e.*, genes in the first half of the chromosome are binary values imposing the constraint $x_i \in \{0, 1\}$ and genes in the second half of the chromosome can only have values imposing the condition $y_i \in J$, where J is the set of possible mating angles.

The constraint on the number of components is imposed as a penalty term in the fitness function by taking the difference of the resulting number of components and the one specified by the user.

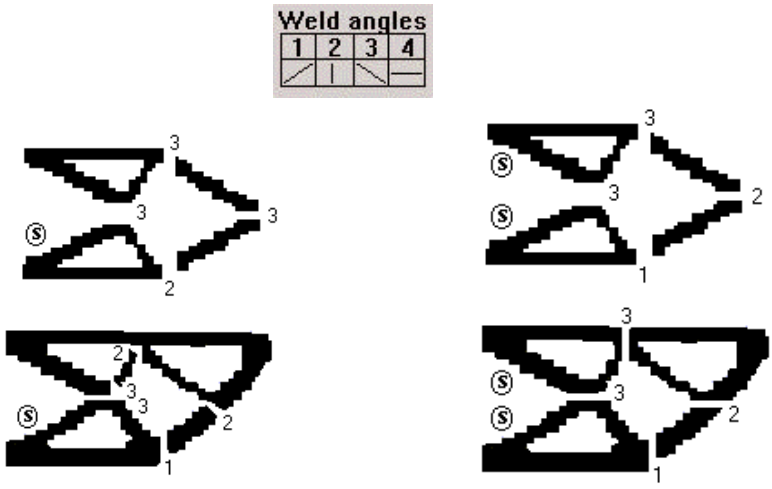
$$\text{Penalty} = (\text{COMPONENTS}(\text{GRAPH}(\boldsymbol{x})) - k)^2 \tag{3.8}$$

Connectivity constraint is implemented by returning a fitness of infinity (very large number in the software implementation) for decompositions lacking connectivity, *i.e.*, returning 0 when passed to the IS_CONNECTED(G) function. Structurally disconnected decompositions, which are not feasible, are eliminated by this constraint straight away.

3.3.2 Decomposition of the products

Using a population ranging between 200 and 300 members, and running the genetic algorithm with a termination condition of about 5000 iterations, several local minima, *i.e.*, optimal decompositions are obtained (Figure 3.10). It turns out that using a small number of iterations is not enough for the system to reach a steady population. As expected, the search space is really spacious and the convergence to a different solution is highly dependent on the random initial population. Fortunately, due to the fact that the finite element analysis is performed *a priori*, and results are stored in a look-up table, the whole process takes around 30 seconds on a Pentium III 800 MHz computer. The fast approximate isomorphism check also contributes to the good speed of the software. So the optimization can be performed repeatedly to cover increasingly bigger areas in the search space, to ultimately reach the global optimum.

The decomposition given in Figure 3.10 (a) is found to be the best solution when modularity consideration has a sufficient weight to force the designs to share a component at all times. Though the best solution agrees with the human intuition that the triangular components in the both products should be shared in some way, note that the ideal case that involves two shared components (Figure 3.10 (b)) has a cost nearly 50% more than the best cost in Figure 3.10 (a). So the expected ideal configuration is essentially not feasible unless the modularity measure is far more important than the structural strength and assemblability considerations. However if some manufacturability criterion was present as well, the best solution might be disregarded due to the complex shape of the second configuration.



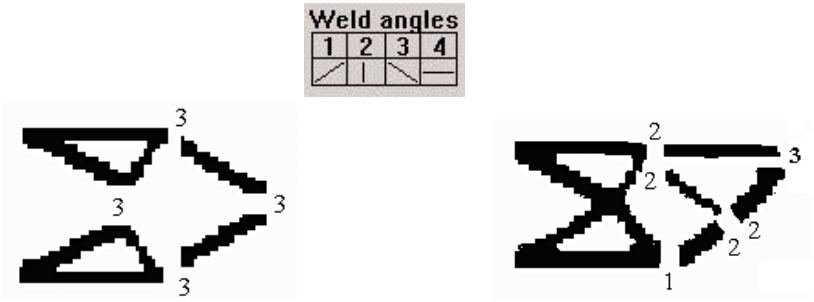
(a) Best solution, cost = 9400 (b) Local minimum, cost = 14,000

Figure 3.10. The minima for the sample problem. The components marked with 's' are shared among the products.

An important observation is that, in the modularity criteria, the term that contains the resemblance of angles cannot be made too large, *i.e.,* the corresponding weight has an upper bound. When one tries to increase this weight to force that the shared components have similar angles, the solutions tend to avoid having part commonality. A further analysis is certainly necessary to investigate this conflict, but at this stage it will be only inferred that practically it is difficult to make the shared components have similar weld angles.

To examine the effects of the modularity terms in the objective function, the earlier version of the assembly synthesis implementation as reported in (Saitou and Yetis, 2000) is used, and the configurations that result from solely structural measures are presented in Figure 3.11. Note that while the optimal configuration in Figure 3.11 (a) agrees with the minima found in the scope of the modularity analysis, the structural measures, when applied alone, lead to different decompositions for the second problem as can be observed by comparing Figure 3.10 with Figure 3.11 (b). For further insight into

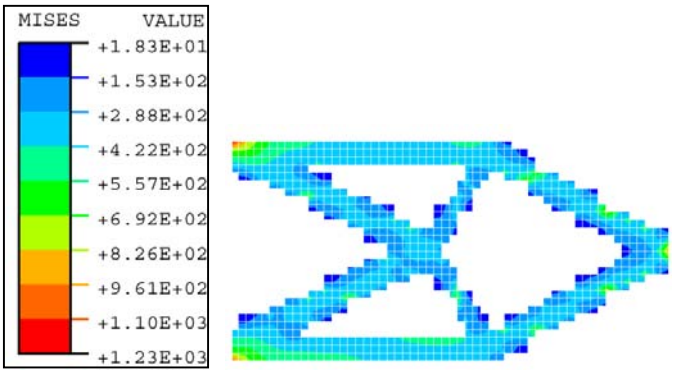
the obtained solution, the ideal angles for the joints and the von Mises stress distributions for both structures are given in Figure 3.12 and Figure 3.13 respectively.



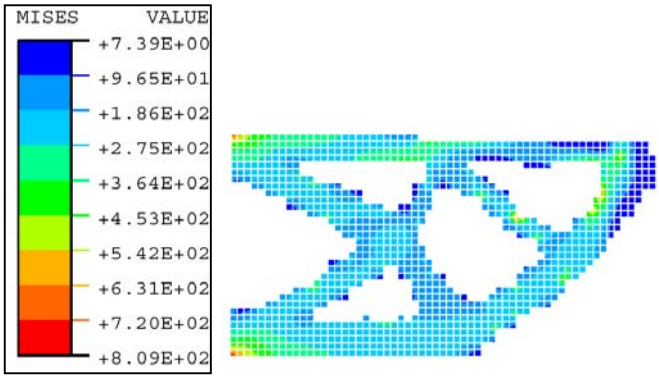
(a) Problem 1, in 4 components (b) Problem 2, in 5 components

Figure 3.11. Decompositions when only structural measures are used.

Figure 3.12. Ideal angles calculated for the joints using the results of the finite element analysis.



a) von Mises stress distribution for structure 1



b) von Mises stress distribution for structure 2

Figure 3.13. Stress distributions for the sample problem. Maximum stresses are 600 MPa and 500 MPa respectively for the two structures. Stress increases from dark to light regions.

3.4 Case Study: Bicycle Frame Design

To demonstrate the capabilities of the method better, a real life problem is addressed as a case study. The goal in this problem is to decompose two similar bicycle frames in an optimal way. Figure 3.14 illustrates the bicycle frame design model as given by Chirehdast *et al* (1994); with a few simplifications in the loading conditions, this model is the starting point of our example. Chirehdast *et al.* use a three-phase design

process called Integrated Structural Optimization System (ISOS). In Phase I, an optimal initial topology is created by a homogenization method as a gray-scale image. In Phase II, the image is transformed to a smoother and realizable design using computer vision techniques. In Phase III, the design is parameterized and conventional size and shape optimization methods are employed. The model in Figure 3.14 is the starting point of Phase I in ISOS environment.

Figure 3.15 presents the original loadings and the resulting optimal topology. Changing the application point of the load on the handle, taking into account the fact that some bicycle models have the handle lower than the seat, the optimal configuration given in Figure 3.16 is obtained. The resulting structures are similar at first sight, but slightly different in geometry, which make them ideal candidates for a modularity analysis. Note that Figure 3.16 (b) is the same as the optimal configuration given by Chirehdast *et al*. at the end of Phase II, and also very close to the customary frames offered by the industry.

The von Mises stress distribution and the ideal angles as given in Figure 3.17 and Figure 3.18 are obtained using Abaqus, with a mesh size of 25 mm by 25 mm. Table 3.1 gives the typical GA parameters used during the optimization run. The weight values for the objective function terms are also tabulated (Table 3.2). To better visualize the influence of the modularity terms in the objective function, the software is run initially with only structural criteria, and the decompositions presented in Figure 3.19 and Figure 3.20 are found to be the optimal solutions. Consequently carrying out the modularity analysis, several local optimal solutions with close costs are achieved (Figure 3.21). The evolution history graph depicting the change in each objective function term and visualizing the convergence of the GA is given in Figure 3.22.

Note that in general the resulting decompositions are similar to the analysis for structural criteria, but especially the second frame converges to a single solution to share the triangular component on the left of the structure. Another interesting observation is the shared component shown in Figure 3.21(a); this solution would not be possible

without the area-moment calculation, which evaluates geometric similarity in a rotationally invariant way.

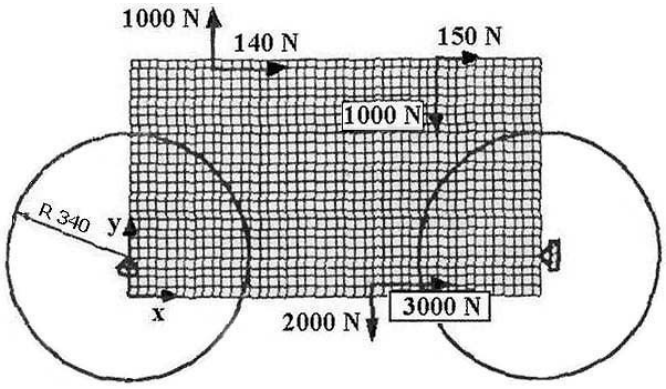
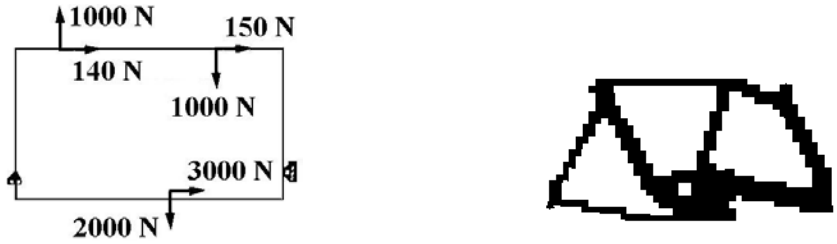
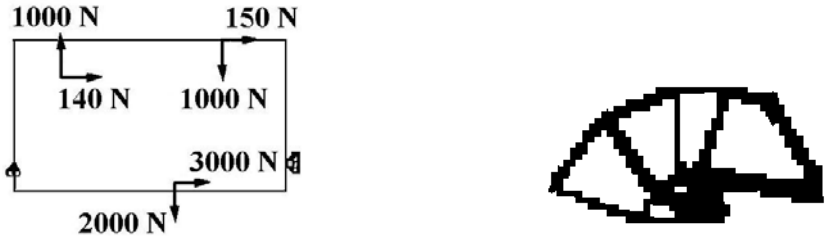


Figure 3.14. Bicycle frame model modified from (Chirehdast *et al.*, 1993).



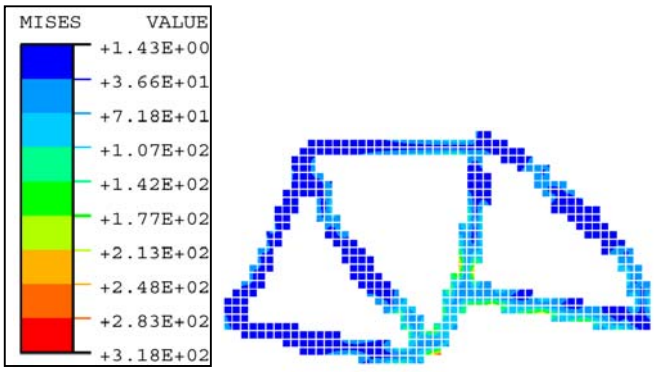
a) Loading for frame 1 b) Resulting optimal topology

Figure 3.15. Loads and boundary conditions for the first frame.

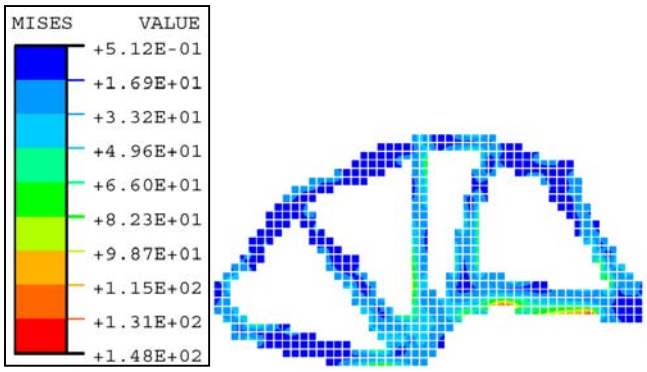


a) Loading for frame 2 b) Resulting optimal topology

Figure 3.16. Loads and boundary conditions for the second frame. Application point of the (1000 N, 140 N)load is slightly changed compared to Figure 3.15.



(a) von Mises stress distribution for frame 1



(b) von Mises stress distribution for frame 2

Figure 3.17. Stress distributions for the frames. Maximum stresses are 170 MPa and 100 MPa respectively for the two structures. Stress increases from dark to light regions.

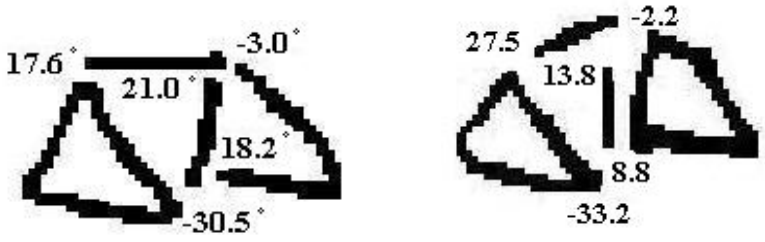


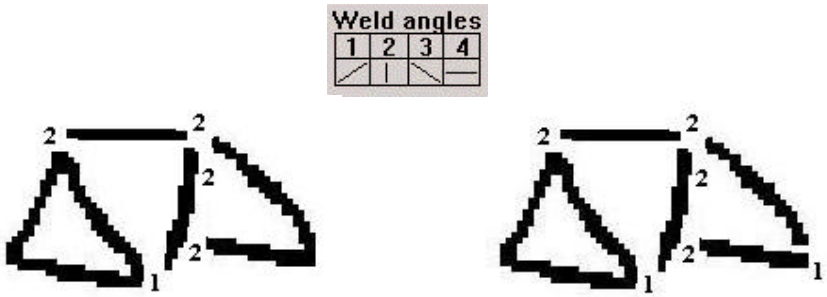
Figure 3.18. Ideal joint angles for the frames.

Table 3.1. Typical run-time GA parameters used in the case study.

Population size	200
Number of generations	3000
Crossover probability	90%
Mutation probability	1%
Population replacement	20%

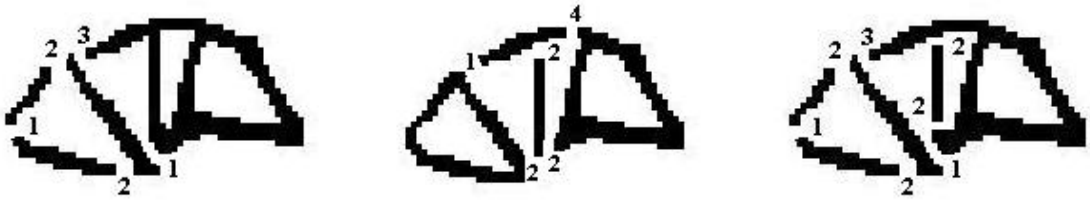
Table 3.2. Typical weight values for objective function terms.

w_1 (deviation from ideal angle)	100.0
w_2 (tensile force on welds)	1.0
w_3 (similarity of joint angles)	10.0
w_4 (number of welds)	1000.0
w_5 (weld angles of modules)	1000.0
w_6 (geometric similarity of modules)	1.0



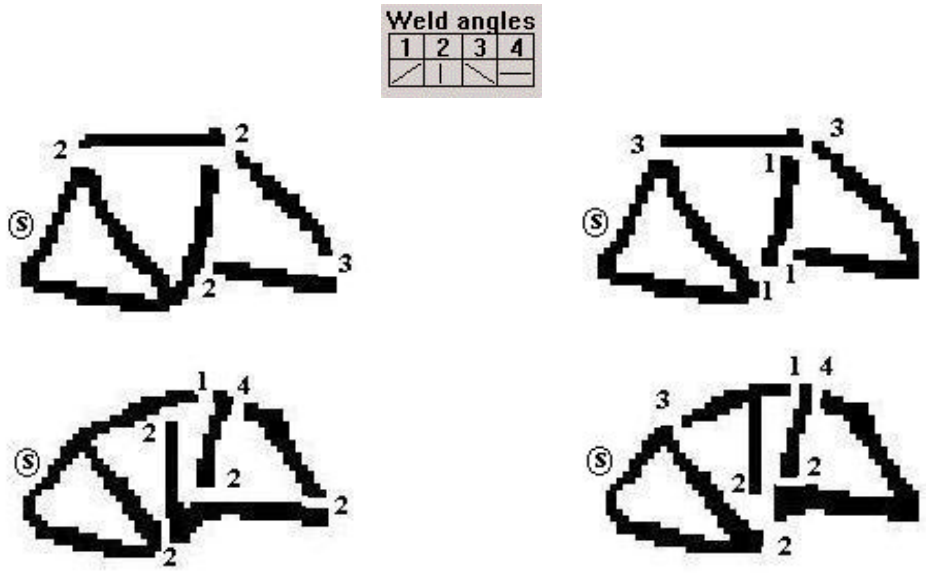
a) 4 components b) 5 components

Figure 3.19. Decomposition of frame 1 when only structural measures are used, i.e. $w_5=w_6=0$.



a) 4 components b) 4 components c) 5 components

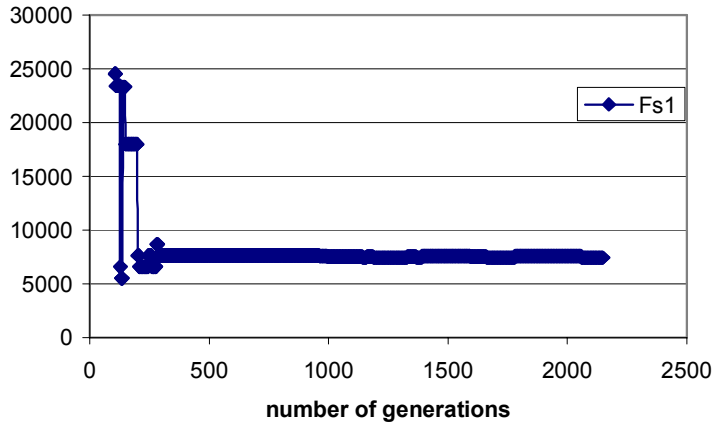
Figure 3.20. Decomposition of frame 2 when only structural measures are used. a) and b) are two alternative decompositions with close cost values.

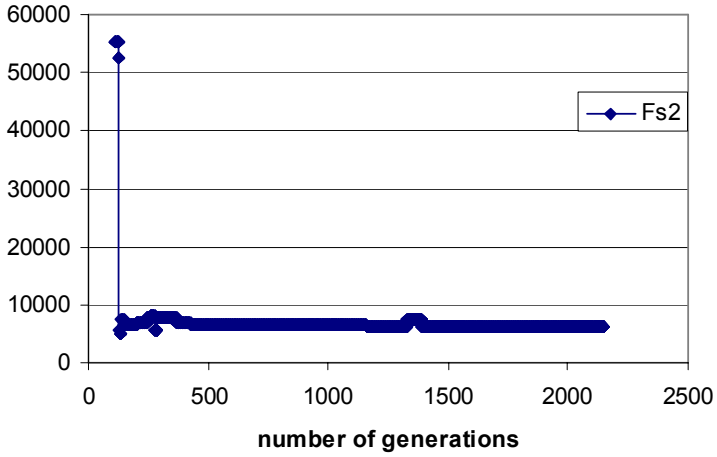


a) 4 components, cost = 15,300 b) 4 components, cost = 14,800

c) 4 components, cost = 13,300 d) 5 components, cost = 18,600

Figure 3.21. Decomposition of the frames for modularity. The components marked with 's' are shared among the products.





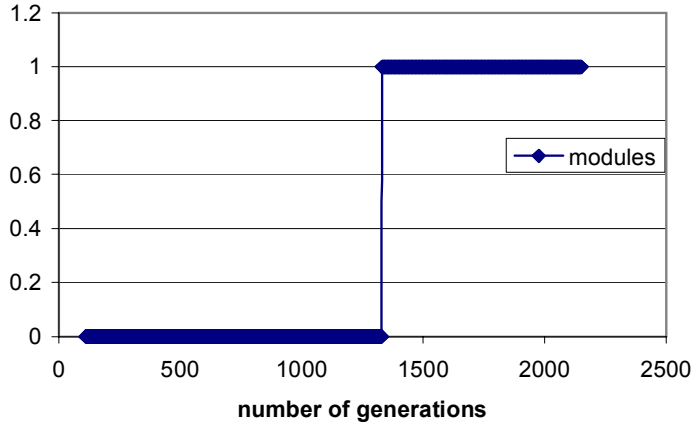


Figure 3.22. Optimization history of a typical GA run for the bicycle design example for each objective function term: $f_s(x_1,y_1)$, $f_s(x_2,y_2)$ and number of modules. The values shown in the plots are of the best individual for each generation.

It is observed during this case study that there are some essential deficiencies in the assembly synthesis method that have to be addressed before more complex applications could be considered:

- Joint models are subject to improvement: design for weld angles is hardly applicable to real-world problems, especially when weld planes are not variables but are completely defined from the beginning by the orientations of the parts in the given structures.
- The design process is unnecessarily sensitive to human input: number of desired components *(k)* is hard to know, and ideally some manufacturing criteria should determine the complexity of the components and consequently the value of *k*.
- It is believed that 3D extension of the method is needed for real-life applications and beam-based models of structures would be a natural next step in this direction.

CHAPTER 4

ASSEMBLY SYNTHESIS FOR BEAM-BASED PRODUCTS

4.1 Introduction

The modular structural component design problem is updated for beam based products in this chapter. The problem is again posed as an optimization of the locations of joints and joint types within two variants of a structural product. Considering automotive body applications, the locations and types of joints are selected to 1) minimize the reduction of structural strength due to the introduction of spot-weld joints in each structure, 2) minimize the number of redundant joint in each structure, 3) maximize the manufacturability of the components via stamping processes in each structure, and 4) maximize component sharing between two structures. The procedure involves the customary two steps of assembly synthesis:

- **Construction of the product topology graphs of each structure:** The designer defines the basic "atomic" components (minimum units subject to decomposition) on each structure and a graph is constructed that represents the connectivity of these basic components within the structure, where each node indicates a basic component and each edge indicates potential joining points. If the basic components are simply defined as the beam segments in a structure (as in the case of the following

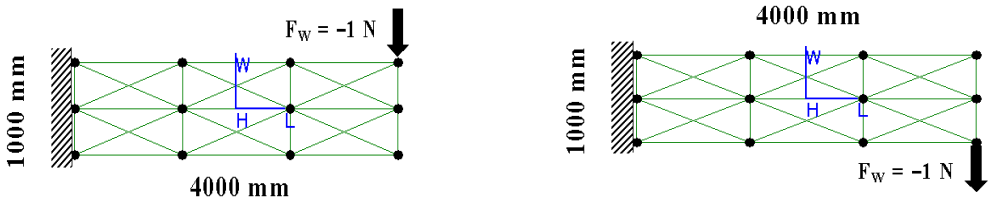
examples), the edges in the product topology graph represents the intersection points of these beam segments[iii].

- **Decomposition of the product topology graphs:** The product topology graphs are decomposed so as to maximize or minimize an objective function while satisfying constraints. In the present method, the objective function is a weighted sum of the functions measuring the structural strength of and number of joints in the assembled structures, the manufacturability of components in each structure, and the amount of component sharing between two structures.

The flowchart for beam-based assembly synthesis is similar to the one given in Chapter 3 (Figure 3.1); actually the method gets simpler by the elimination of image processing routines. Note that beam-based models have an inherent topology segmentation and topology graph can be directly generated using the input that defines the configuration of the structure. Also the input to Step 1 is considered an arbitrary beam-based structure in this chapter, not necessarily an optimal configuration.

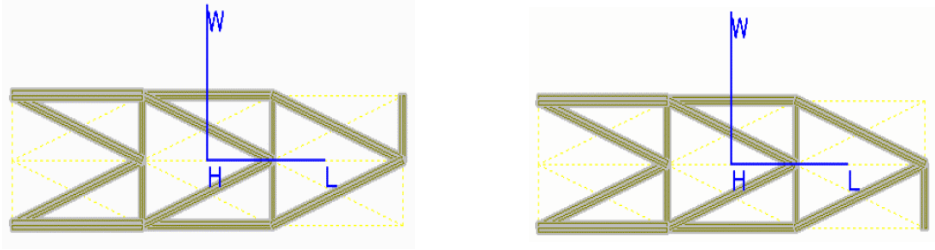
Note that besides the change in the input, there is no difference in terms of formulation and optimization method between 2D continuum-based (as given in Chapter 3) and 2D beam-based applications. Therefore only a simple 2D beam-based example is presented in this chapter. Figure 4.1 and Figure 4.2 demonstrate the optimal decomposition of two cantilevers obtained by a beam-based topology optimization method (First Order Analysis, details are reported in (Cetin *et al.*, 2001)). The design variables are four alternative weld angles (Figure 4.2), similar to Chapter 3.

[iii] If a beam-based structure is seen as a graph, the product topology graph is the dual of the graph of the structure.



Case (1) *Case (2)*

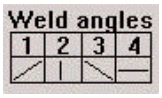
(a) Design domain and boundary conditions



Case (1) *Case (2)*

(b) Optimal designs

Figure 4.1. Design of cantilevers for the base structures for decomposition.



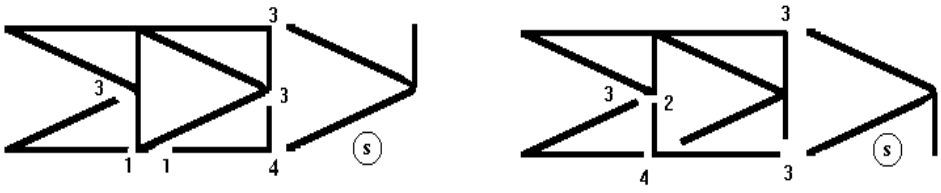


Figure 4.2. Optimal decomposition for the example cantilevers.

There are, however, two important modifications in the formulation for 3D beam-based applications, due to the need to be more realistic and be more efficient parallel to

increasing complexity. First major difference is in the design variables: for 3D structures it is not feasible to design the weld angles at each joint, as this angle is already defined by the orientation of the beams in the 3D space, which is invariant for each configuration. Therefore the new design variable if a weld exists at a joint is the *weld type* and the choices are limited to butt or lap welding, with the alternatives of which beam is to be welded onto other. Another improvement that drives the method towards being realistic is introduction of a manufacturability criterion. For 3D structures as complex as to be handled in the case study of this chapter, the decomposition can not be easily controlled by setting a desired number of components, yet it has to verified that the components are manufacturable. Therefore a new objective function is defined to estimate a cost measure for each component proportional to its overall size and complexity, two major factors that are identified as cost drivers in sheet metal fabrication.

The formulations as wells as the examples in the remaining sections of this chapter are valid for 3D models only. This procedure is also reported in (Cetin and Saitou (b)).

4.2 Mathematical Model

4.2.1 Definition of the design variables

Let a product topology graph be $G = (V, E)$ where V and E are the sets of nodes and edges, respectively. As in the preceding chapters, a decomposition of G into subgraphs can be represented by a $|E|$-dimensional vector $\boldsymbol{x} = (x_1, x_2, \ldots, x_{|E|})$ of a binary variable x_i indicating the presence of edge e_i in the decomposition:

$$x_i = \begin{cases} 1 & \text{if edge } e_i \text{ exists in the decomposition} \\ 0 & \text{otherwise} \end{cases} \tag{4.1}$$

If $x_i = 0$, edge e_i is "cut" in the decomposition and the two components corresponding to the two nodes incident on e_i can be either joined or left as separated. If joined, the type of joints the must be specified. This can be represented as another $|E|$-dimensional vector $\boldsymbol{y} = (y_1, y_2, \ldots, y_{|E|})$ of a variable $y_i \in J$, where J is a set of feasible joint types. Assuming the structure is made of sheet metal with spot weld joints, the four typical types of joints given below are considered (also shown in Figure 4.3):

Type 1: butt joint of beam A onto B (Figure 4.3 (a))

Type 2: butt joint of beam B onto A (Figure 4.3 (b))

Type 3: lap joint of beam A onto B from top (Figure 4.3 (c))

Type 4: lap joint of beam A onto B from bottom (Figure 4.3 (d))

The classification of these types is based on the orientation weld planes that determine the normal and tangential force components the joints are subject to, which is the major governing factor of the joint strength. For this reason, Type 3 (Figure 4.3 (c)) and Type 4 (Figure 4.3 (d)) are distinguished because the weld planes face the opposite directions. It should be noted that beams A and B give a simplification of two strips of bent sheet metals spot-welded to form a closed cross section, as illustrated in more detail in Figure 4.4.

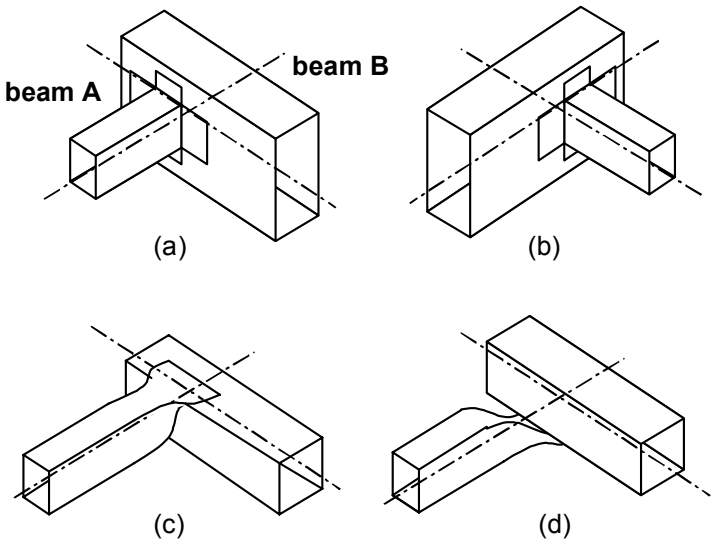


Figure 4.3. Four types of joints that connects two beams A and B. (a) butt joint of A onto B (type 1), (b) butt joint of B onto A (type 2), (c) lap joint of A onto B from top (type 3), and (d) lap joint of B onto A from bottom (type 4).

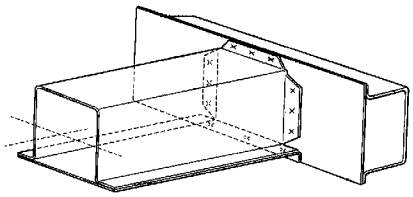


Figure 4.4. A detailed illustration of type 1 joint in Figure 4.3 (a). Beams A and B are also made of sheet metals joined by spot welds.

So the set J consists of four integers representing four different joint types, and 0, denoting the case of no spot weld at the corresponding joint:

$$J = \{0, 1, 2, 3, 4\} \tag{4.2}$$

Note that the value of y_i is ignored when $x_i = 1$ (*i.e.*, no "cut").

4.2.2 Definition of the constraints

The first constraint is on the connectivity of the assembled structures. Since it is possible for an edge e_i in the product topology graph to be cut ($x_i = 0$) and have no weld ($y_i = 0$), a constraint must be in place to ensure the connectivity of the decomposed product topology graphs when assembled back. For both structures this can be expressed in the form:

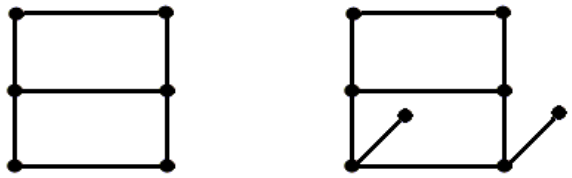
$$\text{CONNECTED}(\text{COMBINED\_GRAPH}(\boldsymbol{x}, \boldsymbol{y})) = \text{TRUE} \tag{4.3}$$

where CONNECTED(G) returns TRUE if the graph G is connected and returns FALSE otherwise, and COMBINED_GRAPH($\boldsymbol{x},\boldsymbol{y}$) returns a graph that consists of the nodes of the original graph and the edges in vectors $\boldsymbol{x}$ and $\boldsymbol{y}$.

The second constraint is on the flatness of the decomposed component to ensure the manufacturability via stamping processes. The flatness of all components in a decomposed product topology graph as specified $\boldsymbol{x}$ can be easily checked geometrically and expressed in the form:

$$\text{FLAT}(\boldsymbol{x}) = \text{TRUE} \tag{4.4}$$

Figure 4.5 illustrates examples of a flat component (manufacturable) and a non-flat component (not manufacturable). The quantitative measure pertaining to the cost of manufacturing each component, namely cost estimation of stamping dies, is included as a part of the objective function.



(a) (b)

Figure 4.5. Example components that are (a) flat (manufacturable), and (b) non-flat (not manufacturable) via stamping processes.

The third and the last constraint guarantees the feasibility of the joint configurations defined in Figure 4.3, which implies that a beam end can be joined onto only one beam. The following function checks this condition for every beam in a structure and returns TRUE if it is satisfied:

$$\text{FEASIBLE\_WELDS}(\boldsymbol{x},\boldsymbol{y}) = \text{TRUE} \tag{4.5}$$

4.2.3 Definition of the objective function

The objective function evaluates a given decomposition as a weighted sum of the following criteria *to be minimized*:

- Reduction of the structural strength in each structure due to the introduction of spot-weld joints.
- Number of redundant joints in assembled structures.
- Manufacturing cost of components via stamping process.
- Dissimilarity of components between two structures.

Since spot weld joints are much less (~5-10 times) strong against tensile loads than against shear loads (Hahn *et al.*, 1997; Radaj, 2000; Davidson), the reduction of the structural strength due to the introduction of spot-weld joints is evaluated as the sum of tensile forces at each joint in a decomposed structure:

$$f_s(\boldsymbol{x}, \boldsymbol{y}) = w_1 \sum_{i=1}^{N_{welds}} \max\{0, \boldsymbol{F}_i \bullet \boldsymbol{n}_i\} \tag{4.6}$$

where w_1 is a weight factor, N_{welds} is the total number of welds in the decomposed structure, $\boldsymbol{F}_i$ is the reaction force at joint i, and $\boldsymbol{n}_i$ is the normal vector of the weld plane pointing to the tensile direction.

The vector $\boldsymbol{n}_i$ is determined by joint type y_i and the angle between joining beams. In the following derivation of $\boldsymbol{n}_i$ for each joint types in Figure 4.3, it is assumed that:

- Only two beams can be joined by a joint, and a joint can have only one weld plane[iv].
- Cross sections of joining beams are rectangular and can be flanged (as in Figure 4.3 (a) and (b)) or flattened (as in Figure 4.3 (c) and (d)) to form a weld plane[v].
- Neutral axes of the two joining beams either intersect each other or are inline.
- Faces of rectangular cross sections of the joining beams are either parallel or perpendicular to the plane defined by the neutral axes of two joining beams.

[iv] While multi-plane joining of two beams can be done in practice, it is not included as possible joint types in Figure 4.3 for simplicity. Inclusion of more joint types is one of the future work.

[v] Note that joint geometry other than the one in Figure 4.3 can realize the same weld plane but it will not make a difference in strength calculation in Equation (4.5).

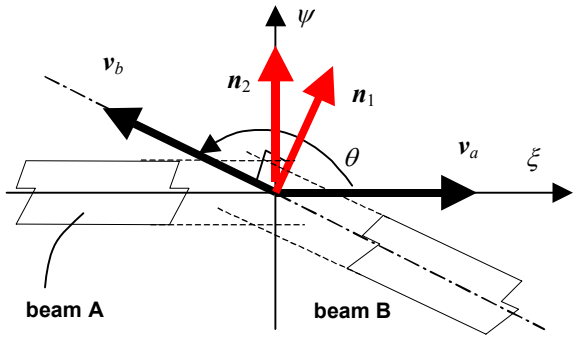


Figure 4.6. Local, right-handed coordinate system ξ–ψ–ζ located at joint *i*, where the origin is at the intersection of the neutral axes of beams A and B, and *x* axis is inline with vector v_a of beam A. Note ζ axis is pointing out of the paper.

Consider a right-handed local coordinate system ($\boldsymbol{o}$, $\boldsymbol{e}_1$, $\boldsymbol{e}_2$, $\boldsymbol{e}_3$) at joint *i*, where $\boldsymbol{o}$ is the origin and $\boldsymbol{e}_1$, $\boldsymbol{e}_2$ and $\boldsymbol{e}_3$ are the bases in ξ, ψ, and ζ directions as shown in Figure 4.6. The origin $\boldsymbol{o}$ is at joint *i*, and ξ axis is inline with vector $\boldsymbol{v}_a$ of beam A (a vector formed by connecting the endpoints of beam A). Note ζ axis is pointing out the paper. Namely,

$$\begin{aligned} \boldsymbol{o} &= \boldsymbol{v}_i \\ \boldsymbol{e}_1 &= \frac{\boldsymbol{v}_a}{\| \boldsymbol{v}_a \|} \\ \boldsymbol{e}_2 &= \boldsymbol{e}_3 \times \boldsymbol{e}_1 \\ \boldsymbol{e}_3 &= \frac{\boldsymbol{v}_a \times \boldsymbol{v}_b}{\| \boldsymbol{v}_a \| \| \boldsymbol{v}_b \| \sin\theta} \end{aligned} \tag{4.7}$$

where $\boldsymbol{v}_i$ is the location of the intersection of the neutral axes of two joining beams A and B, and θ is the angle between two beam as measured in Figure 4.6. Using these notations, normal vector of the weld plain $\boldsymbol{n}_i$ at joint *i* for joint types 1 – 4 in Figure 4.3 are given as:

$$\boldsymbol{n}_i = \begin{cases} \boldsymbol{e}_1 \cos(\theta - 90^\circ) + \boldsymbol{e}_2 \sin(\theta - 90^\circ) & \text{type 1} \\ \boldsymbol{e}_2 & \text{type 2} \\ -\boldsymbol{e}_3 & \text{type 3} \\ \boldsymbol{e}_3 & \text{type 4} \end{cases} \tag{4.8}$$

Note that $\boldsymbol{n}_i$ of type 1 and type 2 are the same if $\theta = 180^o$, *i.e.*, beams A and B are inline.

Since the connectivity of the assembled structure is guaranteed by the constraint in Equation (4.3), the number of redundant welds can be minimized by simply minimizing the total number of welds in an assembled structure:

$$f_W(\boldsymbol{x}) = w_2 N_{welds} \tag{4.9}$$

where w_2 is a weight factor.

In addition to the constraint in Equation (4.4) that ensures the flatness of each component, the cost of component manufacturing via stamping processes is estimated as a tooling cost of stamping die necessary for the component. Since the present method is aimed as a tool during conceptual design phases, only two major factors in the die cost estimation (Boothroyd *et al.*, 1994) are considered in the cost estimation: usable area A_u and basic manufacturing points M_p. The usable area A_u relates to the cost associated with the die size, and computed as the area of the bounding box of a component. The basic manufacturing points M_p is measured by the complexity of stamping die. The empirical data in (Boothroyd *et al.*, 1994) yielded the following second-order polynomial:

$$M_p = -0.0001 X_p^2 + 0.0840 X_p + 30.28 \tag{4.10}$$

where X_p is the die complexity index:

$$X_p = P^2/(LW) \tag{4.11}$$

where P is the perimeter of the component, and L and W are the length and width of the smallest rectangle surrounding the punch, approximated as the bounding box of the component. Figure 4.7 shows the plot of the relationship in Equation (4.10). After all, the manufacturability criteria to discourage complex, large and thus costly parts can be given as:

$$f_c(\boldsymbol{x}) = w_3 A_u^* + w_4 M_p^* \tag{4.12}$$

where w_3 and w_4 are weight factors, A_u^* and M_p^* are the maximum values encountered while examining all decomposed components in a structure.

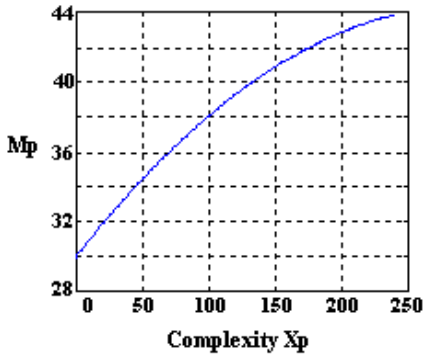


Figure 4.7. Basic manufacturing points M_p vs. die complexity index X_p (Boothroyd *et al.*, 1994).

Let two structures subject to decomposition be structures 1 and 2, and $\boldsymbol{x}_1$ and $\boldsymbol{x}_2$ be binary vectors representing decompositions of structures 1 and 2, respectively. Dissimilarity of components in structures 1 and 2 is evaluated as the negative of the

number of geometrically similar components in the two structures. This can be done by comparing the similarity of each pair of components in structures 1 and 2 as follows:

$f_m(x_1, y_1, x_2, y_2)$

1. module = 0
2. **for** each pair of subgraphs (g_1, g_2) in structures 1 and 2
3. **if** SIMILAR(g_1,y_1,g_2, y_2) = TRUE
4. module = module + 1
4. **if** module = 0
6. **return** a large number
7. **else**
8. **return** -module

where SIMILAR(g_1,y_1,g_2,y_2) is a function that returns TRUE if subgraphs g_1 and g_2 are considered as "similar" both in geometry and in joint types, and returns FALSE otherwise:

SIMILAR(g_1,y_1,g_2,y_2)

1. **if** |AREA_MOMENT(g_1) - AREA_MOMENT(g_2) | < *tol*
2. **and** N_VERTICES(g_1) = N_VERTICES(g_2)
3. **and** ISOMORPHIC(g_1, g_2) = TRUE
4. **for** each matching pair of joints (i, j) in g_1 and g_2
4. **if** ANGLE((y_1)$_i$) != ANGLE((y_2)$_j$)
6. **return** FALSE
7. **return** TRUE
8. **else**
9. **return** FALSE

where *tol* is a given constant, AREA_MOMENT(g), N_VERTICES(g) are functions that return the moment of area with respect to the centroid and the number of vertices, of the convex hull of the component represented by subgraph g, respectively, ISOMORPHIC(g_1, g_2) is a function that returns TRUE if g_1 and g_2 are isomorphic and returns FALSE otherwise, and ANGLE($(y)_i$) is a function that returns the angle of joint i specified by $(y)_i$.

The function ISOMORPHIC(g_1,g_2) is implemented in a generic fashion based on simple node re-labeling (Skiena, 1998), rather than as a theoretically polynomial-time algorithm for planar graphs (Hopcroft and Wong, 1974). This is because the large constant time overhead in the polynomial-time algorithm is not justifiable for the small graphs such as the ones in the present problem. While the current implementation runs in exponential time in the worst case, it practically works fine with the prescreening with the node invariants (Skiena, 1998) such as the degrees of nodes and the lengths of beams corresponding to the nodes.

4.2.4 Formulation of optimization problem

The objective function and constraint described in the previous sections provides the following optimization problem of simultaneously finding modules and module attributes for given two product variants:

- **Given:** structures 1 and 2 and FEM results
- **Find:** modules, joint locations and joint types
- **Constraints:** as given in Section 4.2.2
- **Criteria:** as given in Section 4.2.3

More formally, the problem is formulated as follows:

$$\text{minimize:} \quad f(\boldsymbol{x}_1,\boldsymbol{y}_1,\boldsymbol{x}_2,\boldsymbol{y}_2) = f_s(\boldsymbol{x}_1,\boldsymbol{y}_1) + f_s(\boldsymbol{x}_2,\boldsymbol{y}_2) + f_w(\boldsymbol{x}_1) + f_w(\boldsymbol{x}_2) + f_c(\boldsymbol{x}_1) + f_c(\boldsymbol{x}_2) + f_m(\boldsymbol{x}_1, \boldsymbol{y}_1, \boldsymbol{x}_2, \boldsymbol{y}_2)$$

$$\text{subject to:} \quad \text{CONNECTED}(\text{COMBINED\_GRAPH}(\boldsymbol{x}_1, \boldsymbol{y}_1)) = \text{TRUE}$$

$$\text{CONNECTED}(\text{COMBINED\_GRAPH}(\boldsymbol{x}_2, \boldsymbol{y}_2)) = \text{TRUE}$$

$$\text{FLAT}(\boldsymbol{x}_1) = \text{TRUE}$$

$$\text{FLAT}(\boldsymbol{x}_2) = \text{TRUE}$$

$$\text{FEASIBLE\_WELDS}\ (\boldsymbol{x}_1, \boldsymbol{y}_1) = \text{TRUE}$$

$$\text{FEASIBLE\_WELDS}\ (\boldsymbol{x}_2, \boldsymbol{y}_2) = \text{TRUE}$$

$$\boldsymbol{x}_1 \in \{0, 1\}^{|E_1|}$$

$$\boldsymbol{x}_2 \in \{0, 1\}^{|E_2|}$$

$$\boldsymbol{y}_1 \in \{0, 1, 2, 3, 4\}^{|E_1|}$$

$$\boldsymbol{y}_2 \in \{0, 1, 2, 3, 4\}^{|E_2|}$$

4.3 Optimization Method

As illustrated in Figure 4.8, similar to Chapter 3 and 4, the design variables $\boldsymbol{x}_1$, $\boldsymbol{x}_2$, $\boldsymbol{y}_1$ and $\boldsymbol{y}_2$ are encoded in a "double strand" linear chromosome to preserve the link between $(x)_i$ and $(y)_i$ at joint i during crossover operations.

A software implementation of the optimization problem is done using the C++ programming language with LEDA library developed at the Max-Planck Institute of Computer Science and GALib developed at the MIT CAD Lab. ABAQUS software by Hibbitt, Karlsson & Sorensen, Inc is used for the finite element analyses of the structures.

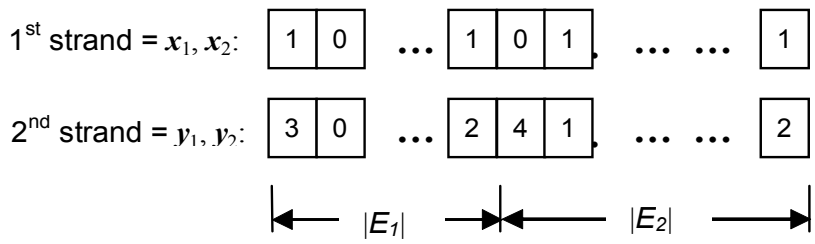


Figure 4.8. Design variables x_1, x_2, y_1 and y_2 encoded as a "double strand" linear chromosome.

4.4 Case Study

This section describes a case study on simplified 3D beam models of a sedan-like body and a wagon-like body shown in Figure 4.9. Both structures are approximately 4.6 [m] in length (x direction), 1.5 [m] in width (y direction), and 1.3 [m] in height (z direction). All beams are modeled as hollow tubes of a 100 [mm] by 100 [mm] rectangular cross section with the wall thickness of 1 [mm]. The material is taken as typical steel with the modulus of elasticity of 200 [GPa].

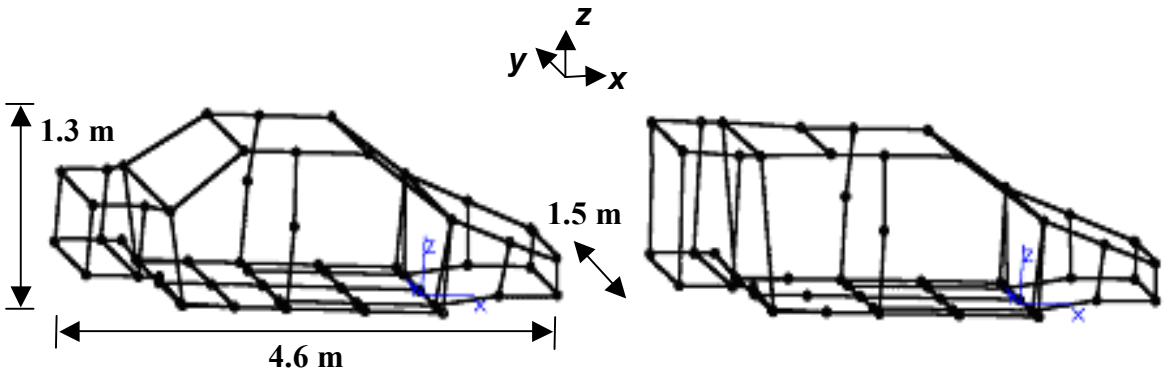
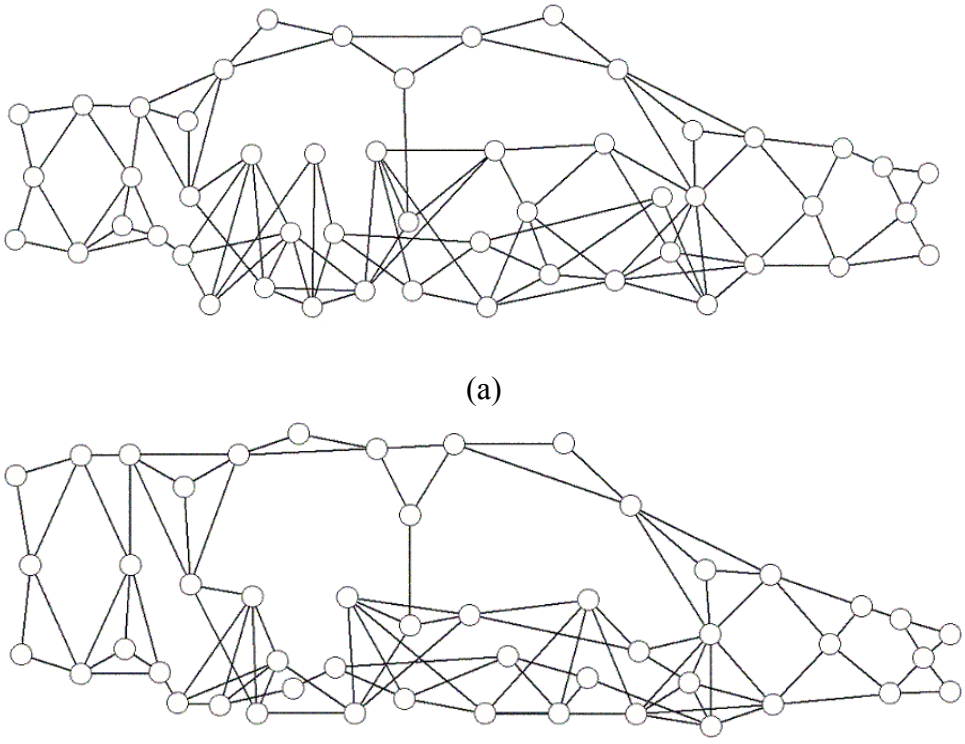


Figure 4.9. Simplified beam-based body structures of sedan (left) and wagon (right). Both structures are approximately 4.6 [m] in length (x direction), 1.5 [m] in width (y direction), and 1.3 [m] in height (z direction).

The decompositions of these structures are conducted under two loading conditions, global bending and global torsion (Malen and Kikuchi, 2002), to illustrate the effects of loading on module designs and overall decompositions. Since the body geometries are symmetric with respect to the x-z plane in Figure 4.9, it is assumed that

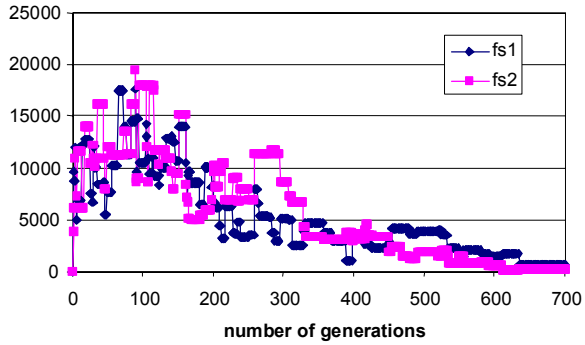
the decomposed components should obey the same symmetry. This allows working on a half of the body during the decomposition processes, reducing the number of variables into a half. Figure 4.10 shows the product topology graph of the sedan and wagon modes, cut in half with respect to the x-z plane. Figure 4.11 shows an optimization history of a typical GA run in the case study, where the values shown in the plots are of the best for each generation, and do not include weight factors. Table 4.1 gives the typical run-time parameters of genetic algorithms used to generate the results in the following subsections.



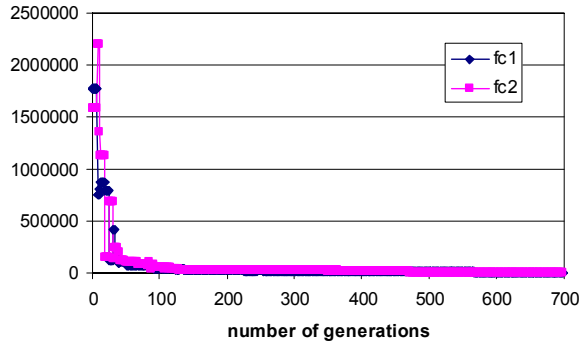
(a)

(b)

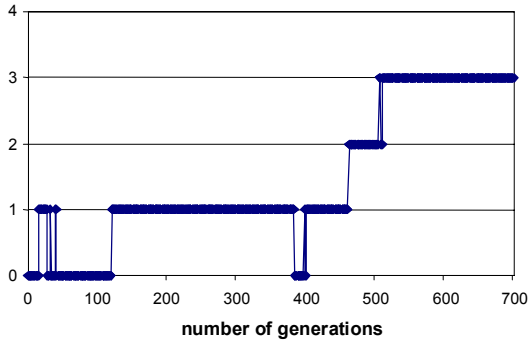
Figure 4.10. Product topology graphs of a half-body with respect to the x-z plane of (a) the structure on the left (sedan), and (b) the structure on the right (wagon) in Figure 4.9.



(a)



(b)



(c)

Figure 4.11. Optimization history of a typical GA run in the case study of (a) $f_s(x_1,y_1)$ and $f_s(x_2,y_2)$, (b) $f_c(x_1)$ and $f_c(x_2)$, and (c) number of modules. The values shown in the plots are of the best for each generation, and do not include weight factors.

Table 4.1. Typical run-time GA parameters used in the case study.

Population size	50
Number of generations	1000
Crossover probability	90%
Mutation probability	1%
Population replacement	30%

Decomposition under global bending

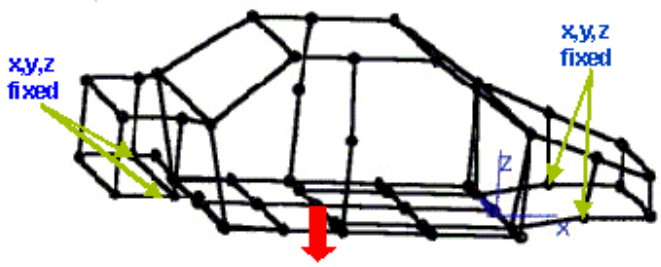
Figure 4.12 shows the boundary conditions of global bending case of sedan and wagon models, where a downward force of 8000 [N] is applied in the middle of the floor as indicated by an arrow. Since the loading is symmetric with respect to x-z plane, half models are also used for the finite element analyses.

Since the component manufacturability is the only criterion that directly favors the decomposition of the structure, its weight in the objective function has a large effect on the number of resulting components. The decomposition results with lower and higher weights in manufacturability criteria are shown in Figures 4.13 and 4.15, respectively. As expected, the number of components increases (or equivalently the sizes of components decrease) as the weight in manufacturability increases. Obviously this result also leads to a larger number of shared modules, as it is much easier for the algorithms to identify simpler components with a higher probability of geometric similarity.

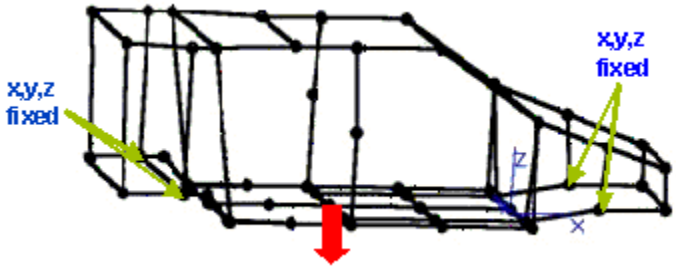
Figures 4.14 and 4.16 show the joint types of the modules in Figure 4.13 and Figure 4.15, respectively. As the formulation punishes tensile forces and shear forces on welds, it can be observed that most of the welds are designed to have lap welds to avoid the almost-pure-shear condition of using butt-welds under global bending (the main load is along z-axis while most members are on x-y plane). Weld types and projected forces on all joints for this loading condition can be found in Appendix A.

Note that determining the locations of the welds, as well as choosing the proper design both strongly affect the overall structural strength of the models. However, the

selection of the location is not driven by projected force calculation only, it is rather subject to intense interactivity with the manufacturing cost evaluation and the manufacturability constraint. Module identification is another factor that attempts to keep some components and corresponding joint designs in place, possibly pushing the design away from the structurally optimal solution.



(a)



(b)

Figure 4.12. Global bending condition on (a) sedan model and (b) wagon model. A downward force of P = 8000 [N] is applied at the location indicated by an arrow.

P

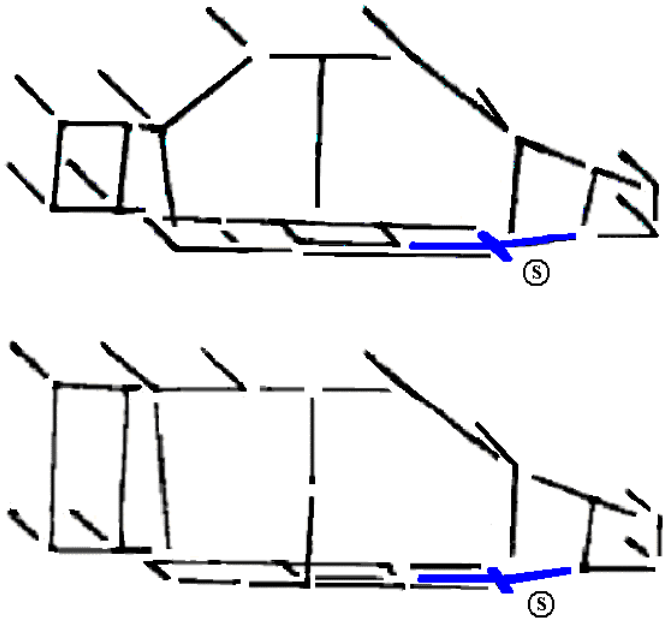


Figure 4.13. Decomposition results for global bending condition with lower weights in manufacturability. The identified module is shown in thick lines and annotated with "s." The number of components in the half sedan body (a) and the half wagon body (b) are 18 and 21, respectively.

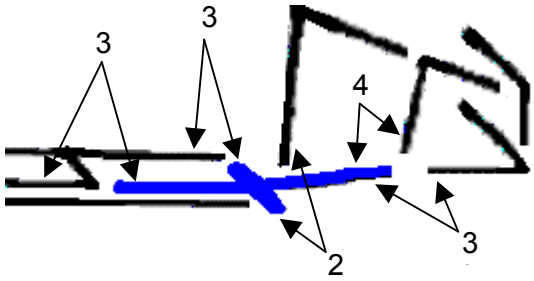


Figure 4.14. Joint types of the module in Figure 4.13. For each joint, beam A in Figure 4.3 is always taken as the beam that is a part of the module.

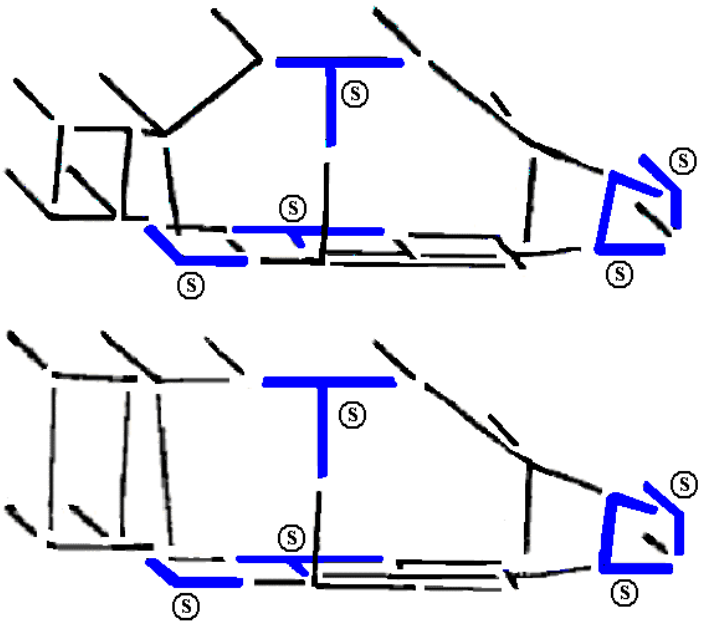
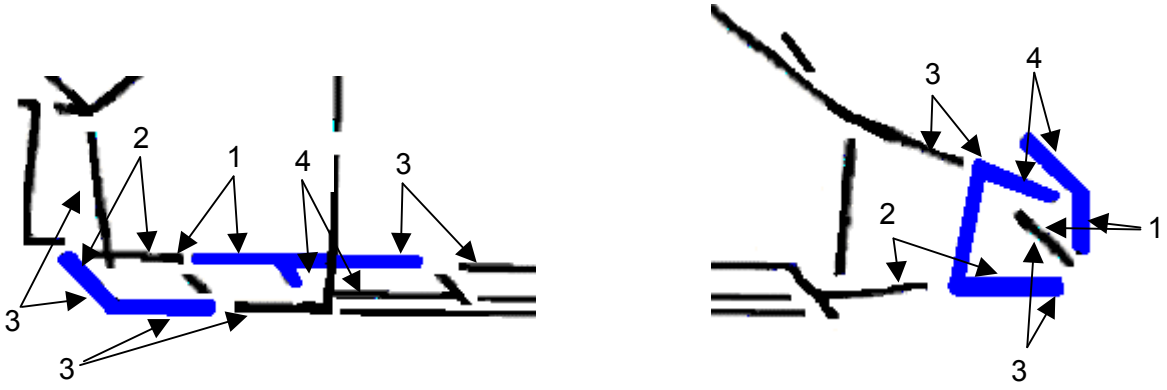


Figure 4.15. Decomposition results for global bending condition with higher weights in manufacturability. The identified modules are shown in thick lines and annotated with "s." The number of components in the half sedan body (a) and the half wagon body (b) are 23 and 26, respectively.

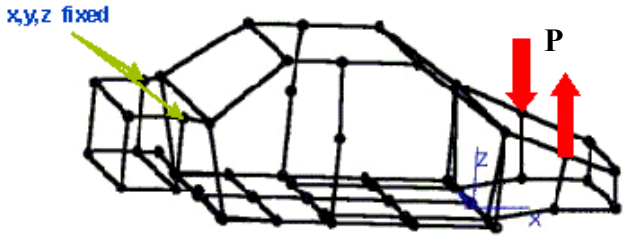


(a) (b)

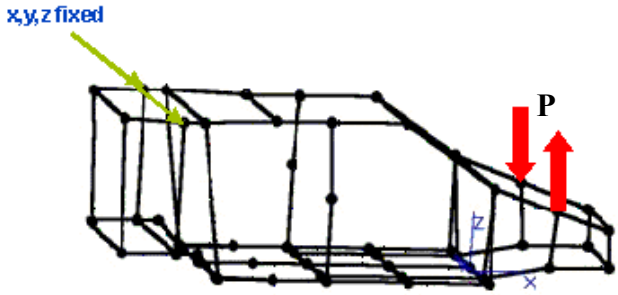
Figure 4.16. Joint types of the modules in Figure 4.15. For each joint, beam A in Figure 4.3 is always taken as the beam that is a part of the module.

Decomposition under global torsion

Figure 4.17 shows the boundary conditions of global torsion case of sedan and wagon models, where upward and downward forces of 4000 [N] each are applied on the sides of front hood, as indicated by arrows.



(a)



(b)

Figure 4.17. Global torsion condition on (a) sedan model and (b) wagon model. Upward and downward forces of P = 4000 [N] are applied at two locations as shown by the arrows.

Note that this loading condition leads to completely opposite force distributions on each side of the x-z plane. As the structural strength criterion in Equation (4.6) is based on the tensile force at each joint, this implies that the best decomposition on one side of symmetry is the worst on the opposite side. In order to identify a decomposition that performs well on both sides rather than the best on one side and the worst on another, tensile force at joint $\boldsymbol{F}_i \bullet \boldsymbol{n}_i$ in Equation (4.6) must be replaced with the worse between both sides:

$$\begin{aligned}\max\{\boldsymbol{F}_i \bullet \boldsymbol{n}_i, \hat{\boldsymbol{F}}_i \bullet \hat{\boldsymbol{n}}_i\} &= \max\{\boldsymbol{F}_i \bullet \boldsymbol{n}_i, -\boldsymbol{F}_i \bullet \boldsymbol{n}_i\} \\ &= | \boldsymbol{F}_i \bullet \boldsymbol{n}_i |\end{aligned} \tag{4.13}$$

where $\hat{\boldsymbol{F}}_i \bullet \hat{\boldsymbol{n}}_i$ is the tensile force at joint *i* on the opposite side of symmetry. Plugging this in Equation (4.6) yields

$$\begin{aligned}f_s(\boldsymbol{x}, \boldsymbol{y}) &= w_1 \sum_{i=1}^{N_{welds}} \max\{0, | \boldsymbol{F}_i \bullet \boldsymbol{n}_i |\} \\ &= w_1 \sum_{i=1}^{N_{welds}} | \boldsymbol{F}_i \bullet \boldsymbol{n}_i |\end{aligned} \tag{4.14}$$

Equation (4.14) is used instead of Equation (4.6) in the decomposition results in Figures 4.18 and 4.20. As in the global bending case, Figure 4.18 shows the decomposition with lower weights in manufacturability and Figure 4.20 shows the one with higher weights. Much larger weights in manufacturability are needed to decompose the structures into the components in sizes comparable to the global bending case. This is due to the fact that Equation (4.14) discourages the decomposition more than Equation (4.6). While for the global bending condition there is a possibility to absorb the compressive forces and avoid the punishment from some force components, for this case it is not possible to ignore negative force values, so the welds become absolutely undesirable. The joint types of the resulting modules in Figures 4.18 and 4.20 are given in Figures 4.19 and 4.21, respectively. Detailed results including weld types and projected forces are presented in Appendix A. It can be observed that lap welds are still much more preferable compared to butt-welding, to keep the (tensile or compressive) forces as low as possible for all joints. It becomes more important under these circumstances to choose the

right joint locations, which is presumably the reason that different decompositions, and consequently different modules, are achieved in the global torsion case.

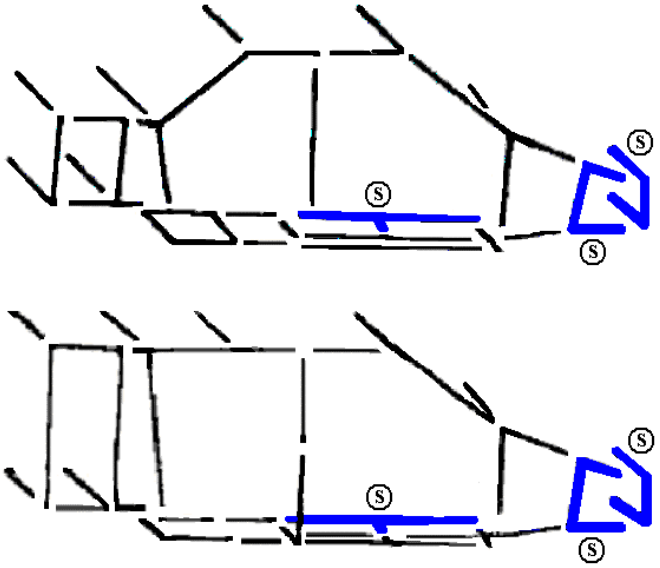


Figure 4.18. Decomposition results for global torsion condition with lower weights in manufacturability. The identified modules are shown in thick lines and annotated with "s." The number of components in the half sedan body (a) and the half wagon body (b) are 21 and 19, respectively.

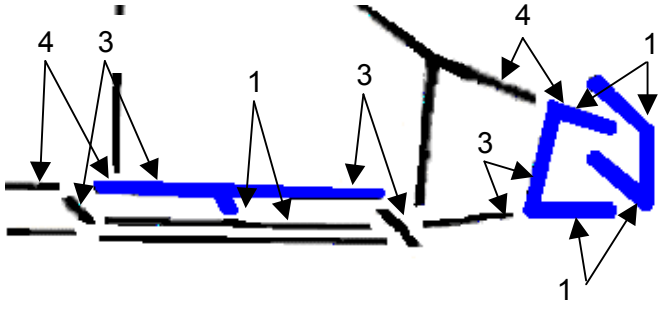


Figure 4.19. Joint types for the modules in Figure 4.18. For each joint, beam A in Figure 4.3 is always taken as the beam that is a part of the module.

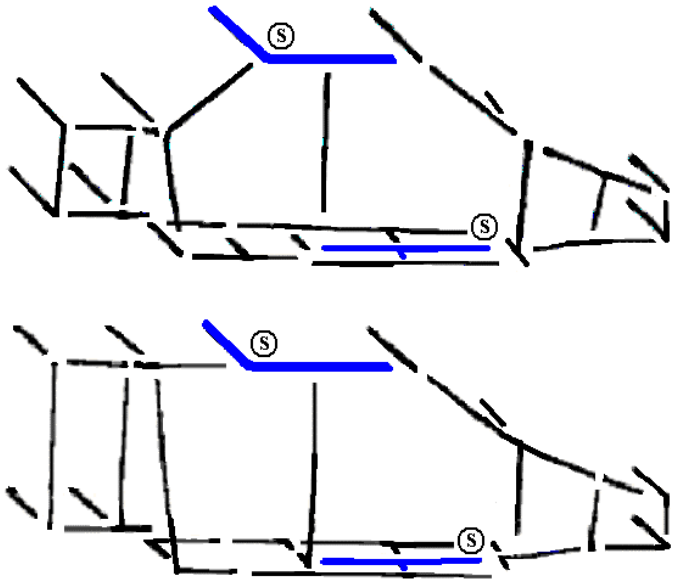


Figure 4.20. Decomposition results for global torsion condition with higher weights in manufacturability. The identified modules are shown in thick lines and annotated with "s." The number of components in the half sedan body (a) and the half wagon body (b) are 25 and 24, respectively.

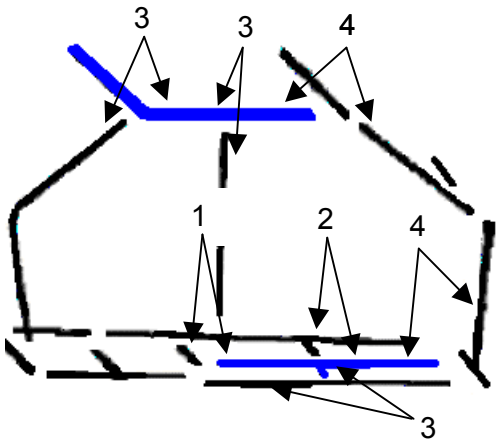


Figure 4.21. Joint types of the modules in Figure 4.20. For each joint, beam A in Figure 4.3 is always taken as the beam that is a part of the module.

This case study proves that many of the shortcomings of the assembly synthesis method as identified in Chapter 3 are eliminated, however there are now some new problems to be solved, mainly because of the increased complexity:

- Design space is believed to be too large as genome representation based on edges leads to many infeasible solutions that are continuously rejected. The method should be modified to explore only the feasible regions in the design space.
- GA does not work as efficiently as desired, as crossover on chromosomes is not able to keep the geometry information of the structures, essential to keep the identified modules in place for newly generated solutions.
- There is still some unnecessary human input in the method: choosing the right weights for the objective function terms requires a lengthy trial-and-error process.
- Module identification is not wise enough: besides the fact that benefit of modularity is not clear, the algorithm is not able to distinguish between sharing simple and complex modules.
- It is believed that application of the method to space frame structures should be a better fit for beam-based representation, and adaptation is not expected to require considerable effort.

CHAPTER 5
ASSEMBLY SYNTHESIS FOR ALUMINUM SPACE FRAMES

5.1 Introduction

Extension of the assembly synthesis method to aluminum space frame (ASF) body architectures (Malen and Kikuchi, 2002) is of strong engineering interest due to their increasing popularity in the automotive industry. Since the space frame bodies are naturally modeled as a network of beams, adaptation is rather straight-forward.

The modular structural component design problem addressed in this chapter is posed as an optimal selection of joint locations and joint types within two ASF structures. The joints are designed to 1) minimize the reduction of structural strength due to the introduction of welds in each structure, 2) minimize the manufacturing costs of two structures considering the opportunities for component sharing under given production volumes.

While overall steps of assembly synthesis are virtually identical to those found in previous chapters, the distinct feature of the present approach is the identification of sharable components as an *outcome* of minimizing the overall production cost, rather than simply maximizing the number of shared components. To quantify the cost reduction of component sharing (assumed to be mainly due to economies of scale), production volumes of both variants are provided as an input to the cost estimation function.

Other improvements introduced in this chapter include a change in the GA representation: the chromosome is modified to cover the possible joints at each physical

node instead of each intersection in the product topology graph. The crossover operator also operates on the physical structure itself instead of the chromosome. After these modifications, the graph generation is kept solely because of the graph isomorphism check that plays an important role in modularity evaluation function.

A final important novelty in this chapter is generation of the Pareto set[vi] as a result of the optimization instead of a single solution, by using a multi-objective genetic algorithm to replace the customary steady-state GA. In this way the designer can compare the relative effects of different design criteria.

The method and the case study given in this chapter are also presented in (Cetin and Saitou, 2003).

5.2 Mathematical Model

5.2.1 Definition of the design variables

In this formulation, the following terminology will be used:

Joint types designate different means of intersection at physical nodes, such as 2-beam intersection and 3-beam intersection, which is invariant in the design and a property of the structure itself. A list of joint types is given in Table 5.1. Figure 5.1 gives the illustrations of all 2-beam joint types, together with an example of a 3-beam joint type.

Joint configurations give the information of which beams are connected to each other to form that joint at the physical node. For each joint type, there are different number of alternative joint configurations, which are the design variables in this problem, given by a set of integers (set J in Table 5.1). Naming the intersecting beams as 1-2-3

[vi] A solution is said to be Pareto optimal if all other solutions have a higher value for at least one of the objective functions, or else have the same value for all objectives.

with respect to their indices, Table 5.2 and 5.3 give all possible configurations for 3-beam intersections, thus the encoding of the discrete sets. While joint types F to H have simply the combinations of beams, type E has two redundant joints eliminated; this is due to the fact that when there are 2 uni-axial beams and a perpendicular one (i.e. forming a 2D plane), there is no difference in welding the perpendicular beam onto first or second uni-axial beams.

Weld types denote the use of alternative welds for each configuration such as butt and lap options (top and bottom illustrations in Figure 5.1 (c) respectively). Weld types are not considered design variables of the genetic algorithm in this study, unlike the previous chapters. As described in next section, the best weld type for each joint configuration is selected based on structural strength.

More formally, for each structure, the design variables at locations $i = 1,2,\ldots N$ are the joint configurations represented as a vector $\boldsymbol{x} = (x_1, x_2, \ldots, x_N)$; $x_i \in J_i$, where J_i is a set of feasible joint configurations corresponding to location (physical node) i and N is the number of nodes in the structure.

Table 5.1. Joint library: depending on the type of the intersection at the physical node, different number of joint configurations exists, as given by a set of integers (*J*).

Joint type	Joint definition	Set J
A	2 beams: uni-axial	{0,1}
B	2 beams: oblique	{0,1}
C	2 beams: perpendicular	{0,1}
D	2 beams: T-joint	{1}
E	3 beams: 2 uni-axial, 1 perpendicular	{1,2,…,7}
F	3 beams: perpendicular in 3D	{1,2,…,9}
G	3 beams: 1 oblique, 2 perpendicular	{1,2,…,9}
H	3 beams: all oblique	{1,2,…,9}

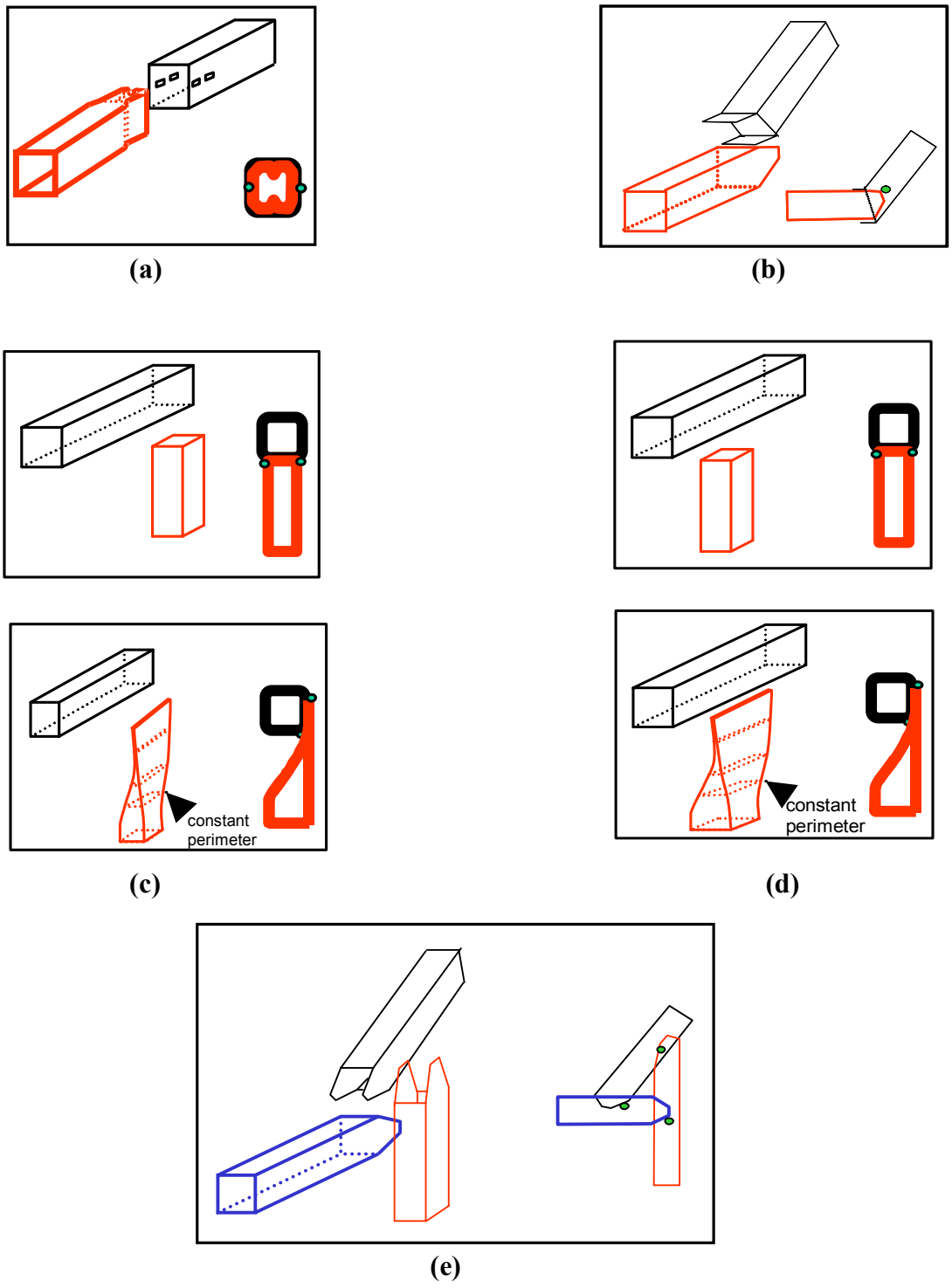


Figure 5.1. Illustrations of joint types: (a) uniaxial (joint type A), (b) oblique (joint type B) (c) perpendicular (joint type C) (d) T-joint (joint type D). It is assumed that 3 beam connections are simply enumeration of these possible joints, an example is given in (e) representing joint type G. These figures are adapted from (Malen and Kikuchi, 2002).

Table 5.2. Details of what the discrete set *J* represents in joint type E; each integer in the set shows a different joint configuration, i.e. alternatives of which beams are welded together or have a solid connection (1-2 uniaxial).

i	J^i *(joint configuration)*
1	(1-2); (3-1)
2	(1-3); (2-3)
3	2-3 solid, (1-2)
4	2-3 solid, (1-3)
5	1-3 solid, (2-1)
6	1-3 solid, (2-3)
7	1-2 solid, (3-1)

Table 5.3. Details of possible joint configurations for joint types F, G and H.

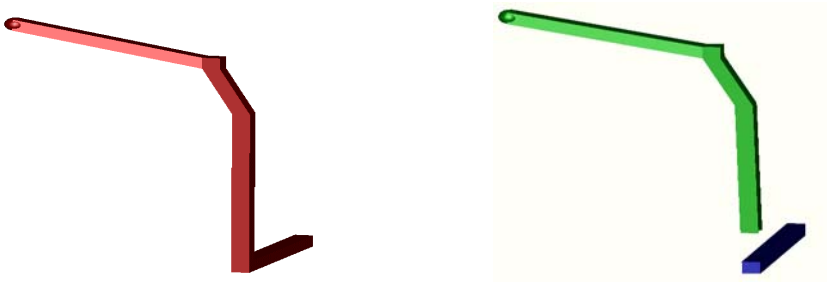
i	J^i *(joint configuration)*
1	(1-2); (3-1)
2	(1-2); (3-2)
3	(1-3); (2-3)
4	2-3 solid, (1-2)
5	2-3 solid, (1-3)
6	1-3 solid, (2-1)
7	1-3 solid, (2-3)
8	1-2 solid, (3-1)
9	1-2 solid, (3-2)

5.2.2 Definition of the constraints

The only constraint in this optimization problem is related to the manufacturability of the resulting components at each decomposition. It is assumed that the only manufacturing operation is bending, so the constraint function **MANUFACTURABLE**($\boldsymbol{x}$) evaluates each component's 'bendability':

$$\textbf{MANUFACTURABLE}(\boldsymbol{x}) = \text{TRUE} \tag{5.1}$$

It is considered that out-of-plane bending is the only infeasible configuration in a component; other complex bending operations are punished by the production cost model, as to be described in Section 5.2.3. Note that a binary constraint that rejects a solution based on the feasibility of a single component is expected to adversely affect the convergence of the optimization run. A solution that is found to work effectively is, instead of using the constraint as a means of rejecting a certain individual (solution), whenever it is possible, it is decided to repair these specific non-manufacturable components. Replacing out-of-plane bends with welds, this binary measure for manufacturability is relaxed to a certain extent and feasible solutions are guaranteed. An example is given in Figure 5.2.



(a) (b)

Figure 5.2. (a) A non-manufacturable component due to the out-of-plane bending requirement, (b) a single-plane manufacturable component that results when a solid connection is replaced with a weld.

The constraint in the previous chapters that is used to guarantee the connectivity of the assembled structures, as well as the function that checks the feasibility of the welds are eliminated in this chapter. Since the chromosome is modified to cover the possible joints at each physical node instead of each intersection, by enumerating the number of different ways the beams at each node could be joined, the connectivity of each physical node is locally guaranteed, eliminating the need to perform an overall connectivity check. Along the same line, the available joint configurations include only the cases that each beam is welded onto a single other beam, making the weld feasibility constraint unnecessary.

5.2.3 Definition of the objective function

The objective function evaluates a given decomposition with respect to the following criteria *to be minimized*:

- Reduction of the structural strength in each structure due to the introduction of welds.
- Production (manufacturing and assembly) cost of components in each structure.

Structural strength criterion

Reduction of the structural strength due to the introduction joints is evaluated as summation of forces on the welds in a decomposed structure.

As explained in the previous section, the design variable vector gives the information of which beams are to be joined (joint configurations), but it does not dictate which weld type is to be used for the connection. As a human designer would simply

choose the strongest weld type if no other criterion is in effect at this point, a straightforward strength comparison of butt and lap configurations is introduced.

In this formulation up to four different weld types are considered: butt and lap, and two alternative orientations (which of the intersecting beams is to be welded onto the other). The classification of these types is based on the orientation of weld planes that determine the normal and tangential force components the joints are subject to, which is the major factor in our definition of structural strength. Exceptions to these four weld alternatives are uni-axial joints, which involve welding one beam inside the other (classified as lap) as in Figure 5.1 (a), and oblique beams, which can only be butt-welded as in Figure 5.1 (b).

The literature on fatigue strength of welds (Ohta and Mawari, 1990; Matsumoto and Izuchi, 1995; Behler *et al.*, 1997; Matthes *et al.*, 1998; Pinho da Cruz *et al.*, 2000; Cederqvist and Reynolds, 2001; Ye and Moan, 2002; also summarized in Appendix B) does not suggest an obvious preference towards evaluation with respect to shear or tensile strength. So to estimate the force on welds, the magnitude of the reaction force on each weld, or the force except the compressive force component is taken, depending on weld type (summarized in Table 5.4). When a weld is in compressive state, i.e. one beam is pressing against each other, weldment is assumed to absorb most of the force. P is calculated by taking the minimum of the values from weld types I-IV in Table 5.4:

$$P = \min(P^{\mathrm{I}}, P^{\mathrm{II}}, P^{\mathrm{III}}, P^{\mathrm{IV}}) \tag{5.2}$$

The objective function term $f_s(\boldsymbol{x})$ is simply the sum of these non-compressive force values over the entire structure:

$$f_s(\boldsymbol{x}) = \sum_{i=1}^{N_{welds}} P_i \tag{5.3}$$

where N_{welds} is the total number of welds.

Table 5.4. Calculation of the force on welds, depending on the joint and weld type. Subscripts denote with respect to which beam the weld plane is defined.

Weld type	Force on weld (***P***)
I. Butt (1st on 2nd)	If *Uni-axial*: N/A If *Oblique*: N/A If *Perpendicular*: norm(max(0,$\|\boldsymbol{F}^t_1\|$), $\|\boldsymbol{F}^s_1\|$) If *T-joint*: norm(max(0,$\|\boldsymbol{F}^t_1\|$), $\|\boldsymbol{F}^s_1\|$)
II. Butt (2nd on 1st)	If *Uni-axial*: N/A If *Oblique*: N/A If *Perpendicular*: norm(max(0,$\|\boldsymbol{F}^t_2\|$), $\|\boldsymbol{F}^s_2\|$) If *T-joint*: N/A
III. Lap (1st on 2nd)	If *Uni-axial*: $\|\boldsymbol{F}^s_1\|$= $\|\boldsymbol{F}^s_2\|$ If *Oblique*: $\|\boldsymbol{F}^s_1\|$ If *Perpendicular*: norm(max(0,$\|\boldsymbol{F}^t_1\|$), $\|\boldsymbol{F}^s_1\|$) If *T-joint*: norm(max(0,$\|\boldsymbol{F}^t_1\|$), $\|\boldsymbol{F}^s_1\|$)
IV. Lap (2nd on 1st)	If *Uni-axial*: $\|\boldsymbol{F}^s_1\|$= $\|\boldsymbol{F}^s_2\|$ If *Oblique*: $\|\boldsymbol{F}^s_2\|$ If *Perpendicular*: norm(max(0,$\|\boldsymbol{F}^t_2\|$), $\|\boldsymbol{F}^s_2\|$) If *T-joint*: N/A

Production cost criterion

The objective function term that estimates the manufacturing cost for each component in each structure is given below:

$$f_c(\boldsymbol{x}) = \sum_{i=1}^{N_{comps}} c_i^{welds}(q_i) + c_i^{bends}(q_i) \tag{5.4}$$

where $c_i(q_i)$ is a cost function that returns a value based on the number of bends/welds of component *i*, and the production volume q_i, defined by the user. Note that the real quantity of a component is determined after it is found out if the specific component is a shared module; if so, the volume is increased to be the sum of the volumes of both products, potentially leading to a reduced cost of production due to economies of scale.

To decide if two components are eligible to be shared modules, geometric and topological similarity of components in structures 1 and 2 has to evaluated by the modularity function:

MODULARITY (x_1, x_2)

1. For all components in decomposed structures 1 and 2
2. If SIMILAR($comp_1$, $comp_2$) = TRUE
3. Store $comp_1$ = $comp_2$ as a *module*:
   - Copy joint configurations of $comp_1$ to $comp_2$
   - Update cost of $comp_1$ = $comp_2$ with new volume

where SIMILAR($comp_1$, $comp_2$) is a function that returns TRUE if two components are considered as "similar" in geometry, and returns FALSE otherwise:

SIMILAR($comp_1$, $comp_2$)

1. If areas of $comp_1$ and $comp_2$ are close in a given *tolerance*
2. AND number of vertices of $comp_1$ and $comp_2$ are equal

3. AND $comp_1$ and $comp_2$ are topologically equivalent

4. Return TRUE

5. Else

6. Return FALSE

The topological equivalence of two components are determined using an *isomorphism* check on the corresponding topology graphs of the components.

5.2.4 Formulation of the optimization problem

The design variables definition and objective function terms described in the previous section result in the multi-objective problem given below; note that there is no constraint in the formulation.

- **Given:** structures 1 and 2 and FEM results
- **Find:** joint locations and joint types
- **Constraints:** manufacturability, as given in Section 5.2.2
- **Criteria:** structural strength and production cost, as given in Section 5.2.3

More formally:

minimize: $\{ f_s(\boldsymbol{x}_1) + f_s(\boldsymbol{x}_2),\ f_c(\boldsymbol{x}_1) + f_c(\boldsymbol{x}_2) \}$

subject to:

$$\boldsymbol{x}_1 \in J_1^{\,1} \times J_2^{\,1} \times \ldots \times J_{N1}^{\,1}$$

$$\boldsymbol{x}_2 \in J_1^{\,2} \times J_2^{\,2} \times \ldots \times J_{N2}^{\,2}$$

$$\textbf{MANUFACTURABLE}(\boldsymbol{x}_1) = \text{TRUE}$$

$$\textbf{MANUFACTURABLE}(\boldsymbol{x}_2) = \text{TRUE}$$

where J is the vector of the set of discrete variables as given in Table 5.1 for each type of joint in the library and N is number of nodes in the structure.

Note that even though it is not explicit in this formulation, we are still solving a single-stage optimization problem of simultaneously finding modules for given two product variants. The decision of sharing modules is now evaluated in the cost function $f_c(\boldsymbol{x})$.

5.3 Optimization Method

In order to avoid the heuristic weight assignment for each objective function term in the formulation (weighted sum method), a multi-objective GA is used in this work.

The use of GA to solve multi-objective problems has been motivated mainly because of the population-based nature of GAs, which allows the generation of several elements of the *Pareto optimal set* in a single run.

With several objective functions, the notion of 'optimum' changes, as in most multi-objective problems the aim is to find good compromises (or 'trade-offs') rather than a single solution, unlike global optimization. A solution is said to be Pareto optimal if all other solutions have a higher value for at least one of the objective functions, or else have the same value for all objectives. Pareto optimal set is the set of solutions which are *undominated* with respect to all other solutions (Coello *et al.*, 2002).

The multi-objective GA scheme implemented in this study is similar to NSGA-II (Deb *et al.*, 2000) but has some slight modifications in enforcing elitism, where the undominated members are copied into a separate elite population. The pseudo-code is given below:

Modified NSGA-II Algorithm

1. Create a population *P* of *n* chromosomes (an encoded representation of design variables) and evaluate their values of objective functions.

2. Rank each chromosome c in P according to the number of other chromosomes dominating c (rank 0 is Pareto optimal in P). Store the chromosomes with rank 0 into set O. Also, create an empty subpopulation Q.

3. Select two chromosomes c_i and c_j in P with probability proportional to *n-rank*(c_i) and *n-rank*(c_i).

4. Crossover c_i and c_j to generate two new chromosomes c_i' and c_j' with a certain high probability.

5. Mutate c_i' and c_j' with a certain low probability.

6. Evaluate the objective function values of c_i' and c_j' and store them Q. If Q contains less than m new chromosomes, go to 3.

7. Let $P \leftarrow P \cup Q$ and empty Q, rank each chromosome in P and remove m chromosomes with lowest ranks from P.

8. Update set O and increment the generation counter. If the generation counter has reached a pre-specified number, terminate the process and return O. Otherwise go to 3.

With the introduction of the physical node representation, genome formulation is renewed to have a single strand (Figure 5.3), to be using values from discrete set J. Each element of vector $\boldsymbol{x}$ fully defines the joint configuration at that physical node, so there is no need to have a 2nd strand in this case, unlike the representations in the previous chapters.

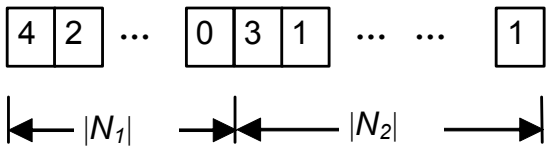


Figure 5.3. Design variables x_1 and x_2 encoded as a single strand linear chromosome. N_1 and N_2 denote the number of nodes in structures 1 and 2 respectively.

To enhance the search efficiency of GA, so-called "direct" or "physical" crossover (Kane and Schoenauer, 1996; Cross *et al.*, 1997; Hobbs *et al.*, 1998; Globus *et al.*, 1999; Fanjoy and Crossley, 2002) scheme is adopted, which directly acts on

phenotype (structures in 3D space in our case) rather than on genotype as the conventional crossovers do. This is achieved by randomly generating a cut-plane in the 3D space, slicing two parent structures with the plane, and then swapping substructures to produce two offsprings:

CROSSOVER(parent 1, parent 2)

1. Choose a random 3D point for the cut-plane, within the bounds of the structure.
2. Generate three random angles to define the direction vector of the normal of the cut-plane.
3. Split both parent structures into two with the cut-plane.
4. Using the properties of the left part of parent 1 and right part of parent 2, form child 1.
5. Using the properties of the right part of parent 1 and left part of parent 2, form child 2.
6. Return child 1 and child 2

As an example, application of the crossover operator on 2D structures is given in Figure 5.4. Here instead of a random cut plane, simply a line is generated to split the structures into two.

Operating on the structures directly has the apparent advantage of keeping the local properties intact, and this is expected to have favorable results in having shared modules in place while looking for better configurations in the rest of the structure. Another opportunity is introducing some bias when selecting the random point on the cut-plane as well as the orientation, so having control over how the structures are to be split. In the context of this paper we are assigning equal probability for every point within the bounding box of each structure.

An optimization case study using aluminum space frames is given in Section 5.4 to further clarify and demonstrate the formulations presented so far.

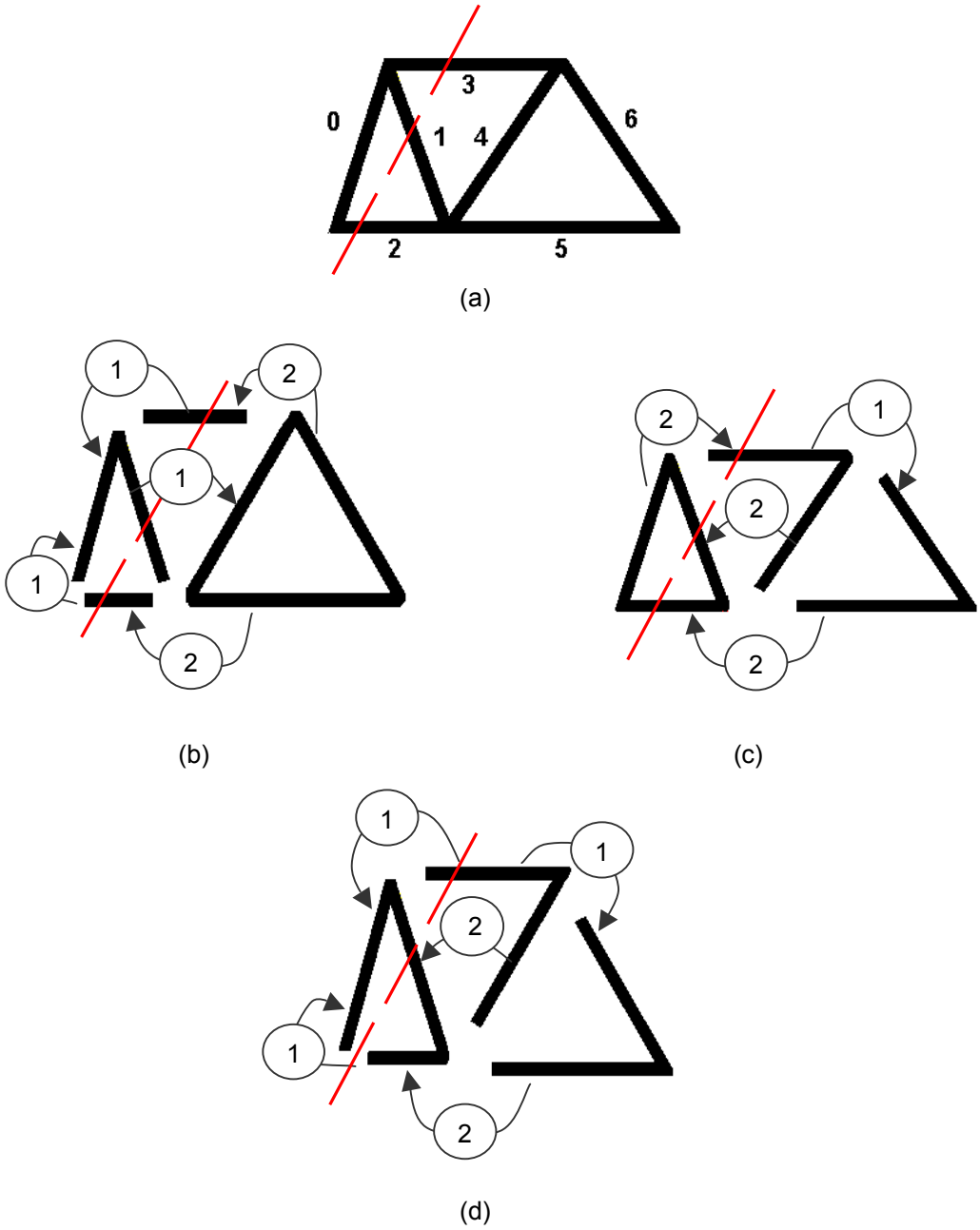


Figure 5.4. Crossover operator directly applied to a 2D structure. A randomly generated line splits the structure given in (a) into two and the child in (d) is formed by taking the joint configurations of the left and right hand side of the line from the parents in (b) and (c) respectively.

5.4 Case Study: Audi A2 and A8 ASF Design

This section describes a case study based on the 3D aluminum space frame models of Audi© A2 and A8 under global-bending loading condition (original models are given in Appendix C). Figure 5.5 shows half of the models used conveniently to simplify the problem, as the structure as well as oading condition is completely symmetric with respect to xy plane. A2 is approximately 3.80 [m] in length (x direction), 1.70 [m] in width (y direction), and 1.55 [m] in height (z direction). A8 is about 5.00 [m] in length, 1.90 [m] in width, and 1.45 [m] in height. Beams are modeled as two types of hollow tubes, 50*50 [mm] or 75*75 [mm] rectangular cross sections with the wall thickness of 2 [mm]. Table C.1 in Appendix C tabulates which cross section is assigned to which beams in the structure. The material is taken as a typical aluminum alloy with the modulus of elasticity of 74 [GPa].

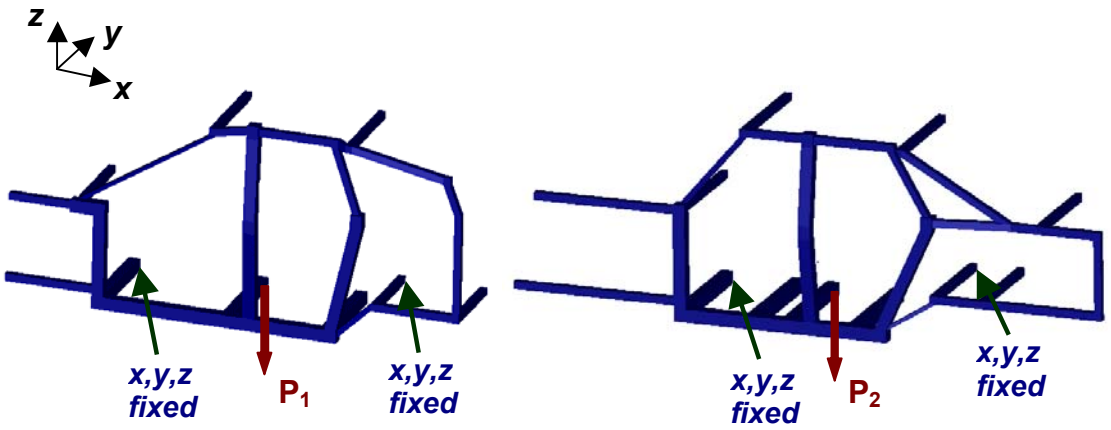


Figure 5.5. Aluminum space frame structures of Audi A2 and A8. Global bending condition: downward forces P_1 and P_2, proportional to the weights of A2 (895 kg) and A8 (1770 kg) are applied at the location indicated by an arrow.

Since the body geometries are symmetric with respect to the x-z plane in Figure 5.5, it is assumed that the decomposed components should obey the same symmetry. So

we will work on the half of the body during the decomposition processes, reducing the number of variables into a half.

The trend in cost decrease given in Figure 2.1 and the data in (Constantine, 2001) and (Clark, 1998) are used to generate the data given in Table 5.5; it is assumed that the only operation to be considered in fabrication is bending when a solid connection is needed[vii] and the only operation in assembly is the welding of the beams. This data is sufficient to carry out the cost estimation given in Equation (5.4).

Table 5.5. Fabrication (bending) and assembly (weld) cost values for each component, adapted from (Constantine, 2001) and (Clark, 1998), respectively.

Production volume	Cost of each weld	Cost of each bend
30	$4.4	$2.9
60	$2.8	$2.4
90	$2.3	$2.1
120	$2.2	$2.0
180	$2.1	$1.9

A software implementation of the optimization problem is done using the C++ programming language. Graph algorithms are developed using the LEDA library from the Max-Planck Institute of Computer Science. ABAQUS software by Hibbitt, Karlsson & Sorensen, Inc is used for the finite element analyses of the structures.

Table 5.6 shows the typical parameters of the multi-objective GA used to generate the results in this section.

[vii] Overall cost of the components is not sensitive to extrusion process; extrusion operation is low-cost, and also much more dependent on the total length of the beams (invariant) rather than the specific decomposition.

Table 5.6. Typical run-time GA parameters used in the case study.

Population size	100
Number of generations	1000
Crossover probability	90%
Mutation probability	1%

For a full analysis of different behaviors under different production strategies, the optimization is run for three different scenarios:

Scenario 1: Both Structure 1 (model based on A2) and Structure 2 (model based on A8) are produced with a volume of 30,000 per annum.

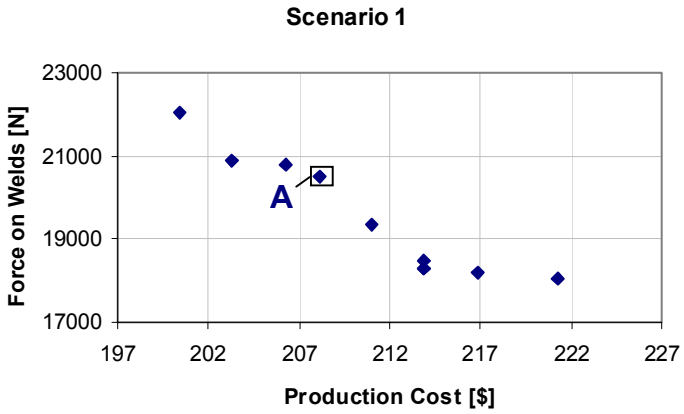
Scenario 2: Structure 1 has a volume of 90,000 while Structure 2 is produced 30,000 per annum.

Scenario 3: Both Structure 1 and Structure 2 are produced as many as 90,000 per annum.

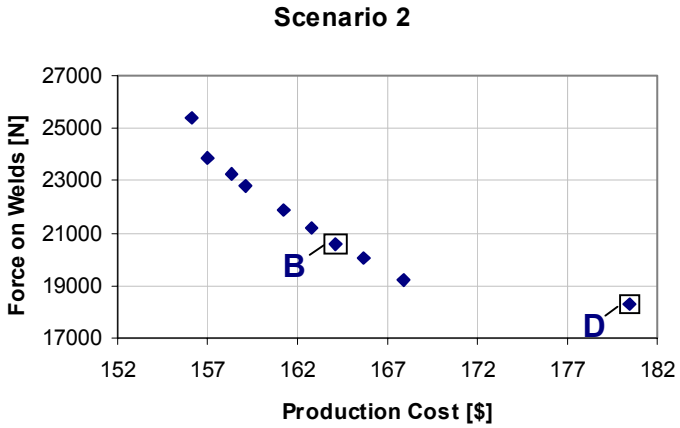
These alternatives are considered sufficient to capture the sensitivity of the algorithm to the different scale economies resulting from different volumes. Running the optimization for each scenario, the resulting Pareto sets given in Figure 5.6 are obtained. An alternative representation, which ranks the solutions with respect to the objective function values is also given in Figure 5.7.

An immediate observation is the expected overall reduction in cost (shift of the Pareto set to left) for increasing production volume, due to scale economies. Scenario 1 has high overall costs in general due to small volumes, but the most important effect is the cost difference between putting a weld and using a bend for this case. For this

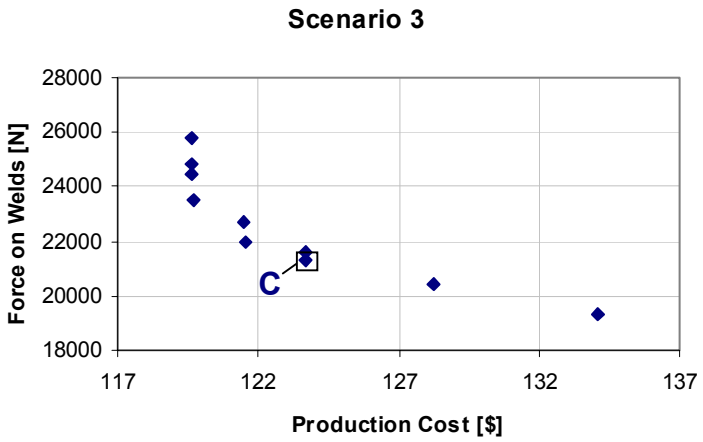
scenario, welds are not desired at all as they are both expensive and cause of a reduction in strength. Thus it is expected that complex components with multiple bends will result from this optimization; only remedy is sharing some modules, effectively reducing the production costs to the level of high volume production. Note that Scenario 3 already has this attribute, a reduced cost per each weld and bend, with values close to each other. Since the scale economies are in effect for this scenario, it is natural not to expect advantages of sharing leading to the results with identified modules. Scenario 2 is mid-way between these two cases, combining high and low volume production for A2 and A8 respectively.



(a)

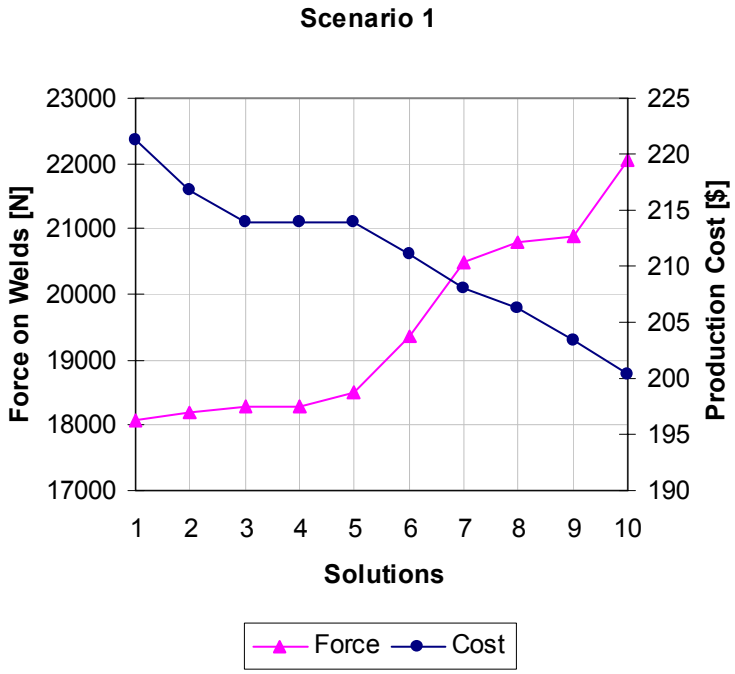


(b)

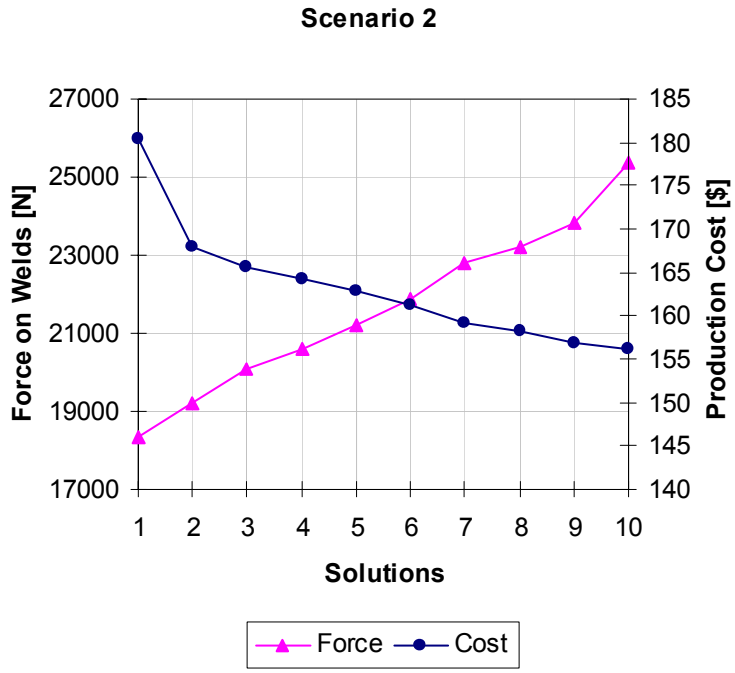


(c)

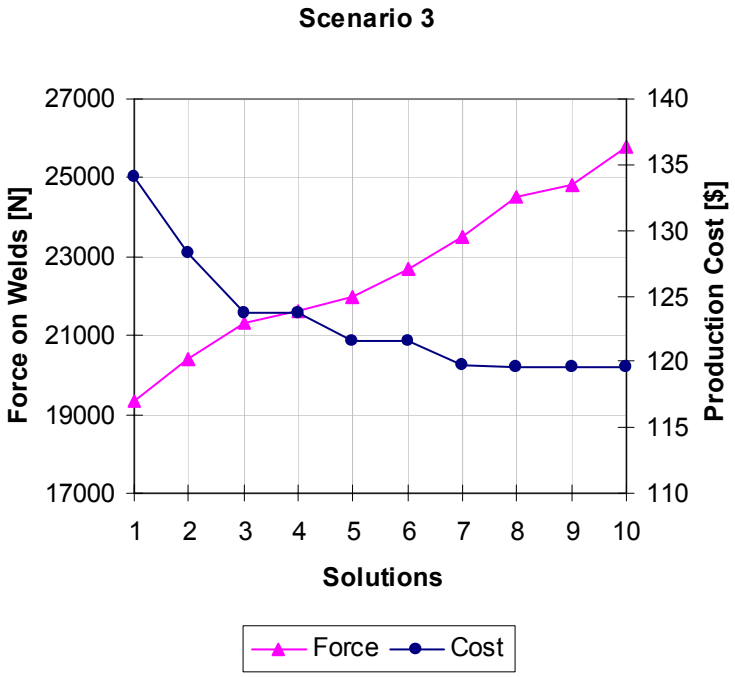
Figure 5.6. Pareto optimal solutions for scenarios 1,2,3 are given in (a), (b) and (c) respectively.



(a)



(b)



(c)

Figure 5.7. Alternative illustration of Pareto optimal solutions for scenarios 1,2,3.

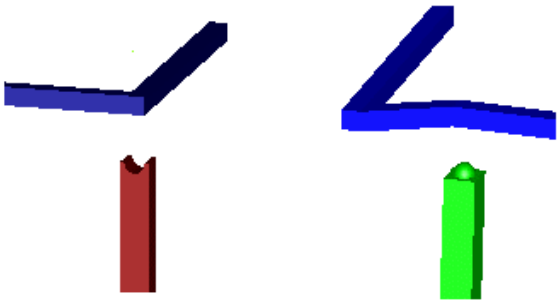
To visualize the major trends, points relatively close to origin for each scenario are chosen and the resulting decompositions are illustrated. Two special notations as given in Figure 5.8 are used to conveniently present the weld types assigned for each joint. Figure 5.9 shows a solution from Scenario 1: this solution attempts to avoid use of welds with some complex components, but the real cost reduction is achieved by identifying two modules that can be shared. Figure 5.10 gives a solution (point B) that still points at two potential sharable components, but the complex module of Scenario 1 is no more recommended. This may be due to changing dynamics in the optimization as the production volumes are different so as the closeness to economies of scale and consequently the drive for modularity. Figure 5.11 samples a point from the Pareto set of Scenario 3, with no modularity at all; this is not surprising as both A2 and A8 are considerably down the cost curve of Figure 2.1 for this scenario. Sharing some components and doubling the production volume do not offer cost reductions any more, so the GA, unable to realize the benefit, stops favoring the solutions with modules.

In most decompositions the uniaxial configurations are observed to be solid, with no bends and welds; a natural solution for low cost production of extruded structures.

Since the two space frame models are not very similar in geometry, the difficulty in finding shared modules is not very surprising. A solution to this is lowering the tolerance used in SIMILAR($comp_1$, $comp_2$) function in modularity evaluation. For this case study this value is decreased to 80% closeness, as opposed to nominal value of 90% used in the preceding chapters.

As the structural strength formulation in this chapter punishes only tensile forces, and butt welds can hardly take advantage of this consideration in case of global bending, similar to Chapter 4, lap welds are the favorable solutions for all decompositions (forces on each joint for Figures 5.9-5.12 are listed in Appendix C). It is also important that since the weld types are no more the variables of the genetic algorithm but are chosen with an inner loop at each iteration, the process of choosing the right location for the joint

becomes more efficient. The algorithm for this case is aware of the weld types with lowest possible forces for each joint configuration, so the joint locations with low reaction forces are found much more quickly.



(a) (b)

Figure 5.8. Illustrations to be used for (a) lap welds (b) butt welds in the visualization of results.

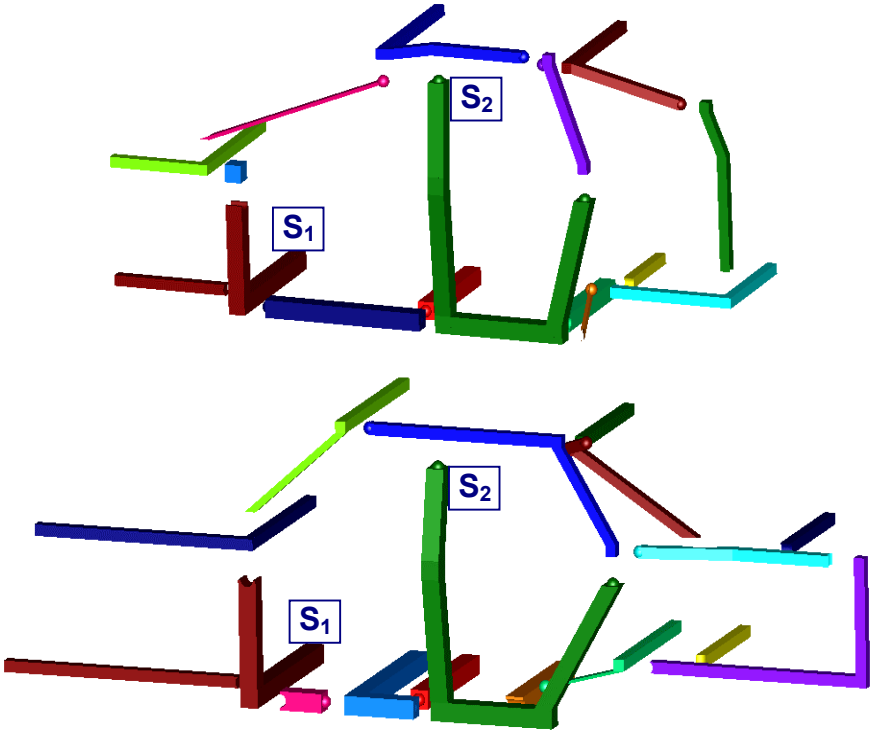


Figure 5.9. Decompositions corresponding to point A in Scenario 1.

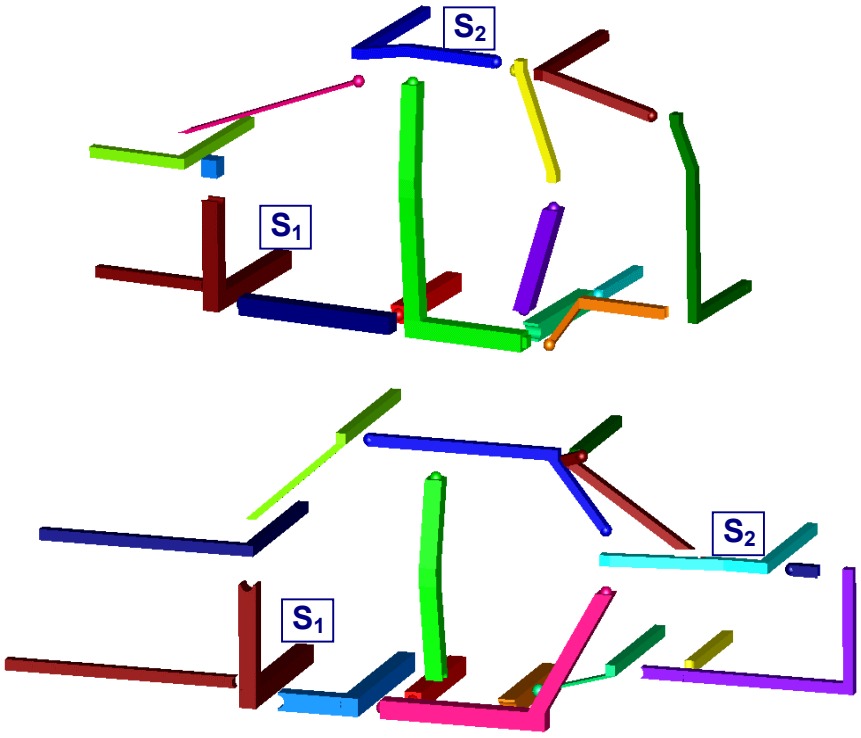


Figure 5.10. Decompositions given by point B in Scenario 2.

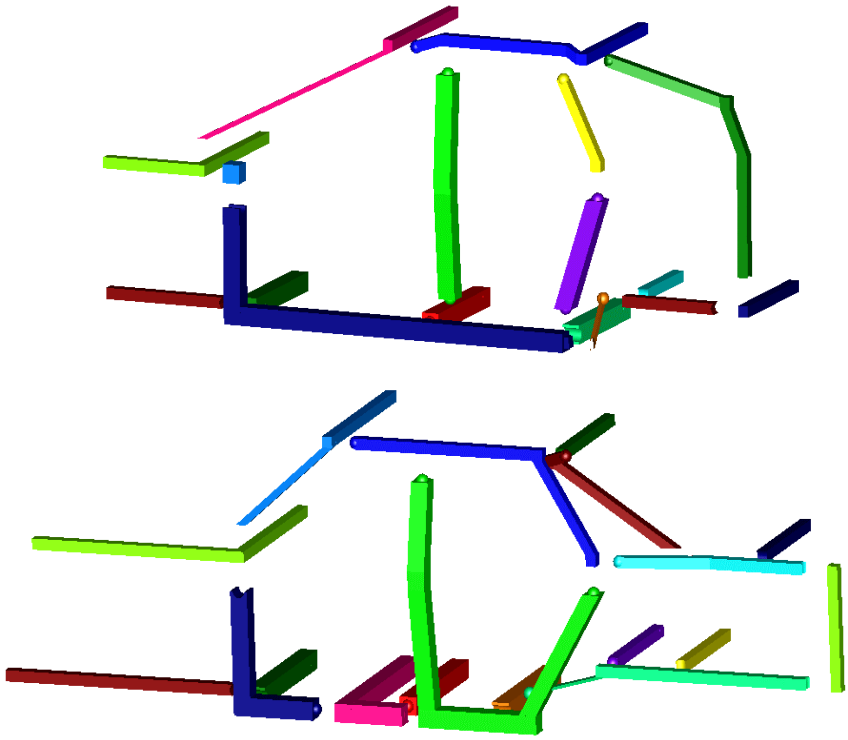


Figure 5.11. Decompositions given by point C in Scenario 3.

Another area of interest for the designer is the dynamics of decomposition and component sharing in the Pareto Set of a certain scenario. It is observed in the solutions of Scenario 1 that in every decomposition two modules that can be seen in Figure 5.9 are identified; different locations and number of welds and resulting complexities of the components determine the cost and structural strength ranking of the Pareto Set. Solutions of Scenario 3 have the same tendency, except that besides possibly random emergence of a few modules, commonality is totally ignored. However, Scenario 2 requires special attention due to the different production volumes of the two models and resulting different sensitivities of the objective functions. Figure 5.12 shows the solution corresponding to point D in Figure 5.6 (b), the strongest but the most costly solution in the Pareto Set. When compared to Figure 5.10 (point B), immediate observations are twofold: decompositions of A2 in both solutions are exactly the same, and there is only one shared component in Figure 5.12. It can be inferred that the GA tends to push the low-volume model (A8 in this case), at the expense of structural strength, to have a similar decomposition to the high-volume model (A2) for some component sharing to occur. This result can be attributed to the fact that low-volume components are costly, thus better candidates for sharing.

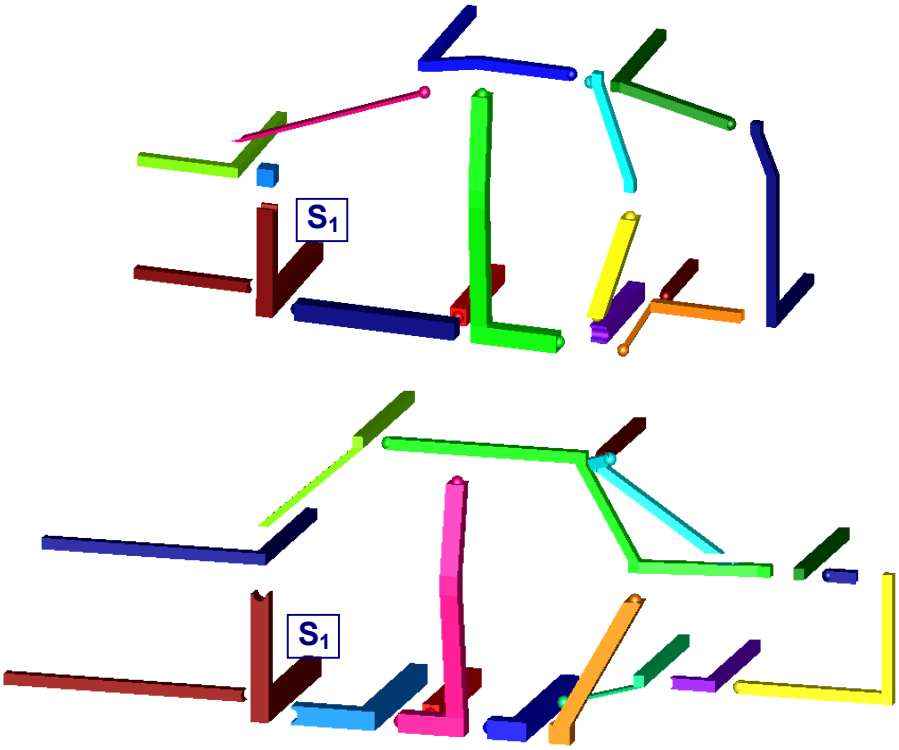


Figure 5.12. Decompositions given by point D in Scenario 2.

CHAPTER 6
CONCLUSION

6.1 Summary and Discussion

The main goal of this thesis is development of an effective method to identify sharable components early in the design process. The proposed approach involves modifying the decomposition-based assembly synthesis method for the design of product families, by identifying modules during the decomposition process and quantifying the trade-offs related to component sharing. The problem is posed as an optimization to minimize the reduction of structural strength due to the introduction of joints while maximizing the manufacturability and assemblability, or equivalently, minimizing the overall production cost.

Section 6.1.1 gives a summary of the thesis and discusses the evolution of the formulations while adapting the assembly synthesis method to beam-based products and 3-D automobile body structures. An overview of the results and conclusions will follow in Section 6.1.2.

6.1.1 Summary

Even though the starting principle of designing the locations and attributes of joints still holds true by the end of Chapter 5, there are several changes in GA representation that is observed to have considerable positive effects on the optimization process. Instead of choosing the right weld angle for each joint, one of the joint configurations that is feasible for that specific joint type is selected during the

optimization, resorting to the joint library specialized for each structural product at hand. The chromosome is then modified to cover the possible joints at each physical node instead of each intersection, by enumerating the number of different ways the beams at each node could be joined. With this representation, the design space gets smaller, as the exploration is limited to feasible regions only; note that even though the discrete variable set for each gene increases, the length of the chromosome decreases drastically to reduce the number of possible solutions. An additional benefit is that, in this way the connectivity of each physical node is locally guaranteed, eliminating the need to perform an overall connectivity check.

Removing the connectivity constraint contributes to the efficient operation of GA, as binary constraints are usually observed to be adversely affecting the convergence of the optimization run. As the manufacturability criteria in 3D applications come in, however, a similar kind of problem happens to arise again. A solution that is found to work effectively is, instead of using the constraint as a means of rejecting a certain individual (solution), whenever it is possible, it is decided to repair these specific non-manufacturable components. Cutting off beams that violates flatness in Chapter 4, and replacing out-of-plane bends with welds in Chapter 5, in this way binary measures for manufacturability were relaxed to a certain extent and feasible solutions are guaranteed.

The manufacturability/assemblability evaluations given through Chapter 3 to 5 are in the form of a set of guidelines for 2D applications and are then replaced with a production cost model that aims to cover all these heuristics. Similarly, the component sharing is imposed as a constraint in the beginning, taking the benefit of modularity as granted, and then incorporated into the cost model to make component sharing an outcome of the optimization for the ASF application. As such constraints are handled very similar to objective function terms in GA applications, this is equivalent to reducing the number of functions in multi-objective optimization. In the end, Chapter 5 introduces a problem with two different objective functions, structural strength and production cost.

This final formulation is expected to sound very natural to the human designer who would be interested to see the trade-off between two criteria totally independent of each other and effectively representing the designer's and management's concerns.

The original assembly synthesis method involved generating topology graphs and posing the search for an optimal decomposition as a graph-partitioning problem. Until the method is modified for ASF applications, this graph-based procedure is followed in this study. In the ASF case, the genetic algorithm operates on the physical structure itself, considerably modifying the optimization process for better operation with modularity considerations. The graph generation is still kept by the end of Chapter 5 though, because of the convenient operation of graph isomorphism check that plays an important role in modularity evaluation function.

The approximation used instead of a formal *graph isomorphism* check in earlier applications seems to be working well, obviously introducing a faster evaluation of the objective function. However it is observed that even in 3D applications, the largest graph encountered in case studies do not have the size that makes the exact check completely impractical, so it is decided to implement an exact isomorphism check. While the current implementation runs in exponential time in the worst case, it practically works fine with the prescreening with the node invariants such as the degrees of nodes and the lengths of beams corresponding to the nodes. This step also makes the matching of joints in two structures very convenient during interface similarity operation, by one-to-one assignment of nodes, which otherwise requires a geometric comparison that may be costly.

Assembly synthesis is unavoidably a multi-objective problem; when the straight-forward weighted sum method is in use in early applications, it is required to carry out the synthesis with different objective function weights in a systematic way to have a complete understanding of the design task at hand. In the case studies the weights are determined by trial-and-error, taking into account the relative function values. However

weights in the objective functions affect the resulting decomposition, hence the identified modules, making it difficult for the user to quickly achieve reliable results. It is therefore ideal to generate the entire Pareto set, so the designer can compare the relative effects of different design criteria. Multi-objective optimization does not deal with global optima, but rather supply a non-dominated set of points, which are to be evaluated by a human decision maker before any further step is taken. Without major computational effort, this is done with a multi-objective genetic algorithm.

Besides the objective function weights, there are two user-specified parameters in the early applications that potentially have influence on the optimization results. One is the shape similarity tolerance that is used to decide if two components are geometrically similar; this parameter may have an effect in some problems, and actually can be used to introduce some robustness into the module identification process, especially when the variant structures are different in size. The second parameter, the desired number of components (k), however, is much more critical. Though as done in some case studies in this thesis, several different optimization runs with different k values are feasible, for complex problems the value could be hard to predict. Ideally the number of components should be the result of the optimization, instead of an input, which is realized when the manufacturability criterion is implemented. The decomposition in this case results in the optimal sizes of components, and consequently the optimal value for k.

6.1.2 Discussion of the results

Either obtained by choosing appropriate weights iteratively in the weighted sum method, or by efficiently generating the Pareto front, the optimization results indicate that there is one outweighing trade-off in this problem: it is between structural strength and manufacturability. Strength deteriorates when excessive number of welds is used, but it is also not a good idea to reduce the number of welds as it will lead to very complex,

difficult manufacturing processes. This fact is very visible in Chapter 4, when the manufacturability criterion is first introduced, which punishes both large and complex parts, but becomes more subtle in Chapter 5, as at this point the welds are assigned cost values as well. Especially for lower production volumes, the cost per each weld is so high that because of their adverse effect on both strength and cost, there is no reason for using welds if not dictated by constraints. However, there are still two factors that drive the populations to the use of welds. First is due to the cost model itself: parts with many bends are discouraged since the production cost is doubled once the total bending angle in one component exceeds 180°, an effective means of reflecting the increasing difficulty in handling complex products (for instance fixturing). The second factor is more indirect, but observable in all case studies covered in this thesis: modularity criterion also has influence on decomposing the products into simpler parts. Note that until Chapter 5, sharing a small, simple component is equally beneficial to sharing a large, complex part, so it is natural that the GA favors increased number of resulting components at each decomposition, which generally increases the potential of geometric similarity. The cost model makes this phenomenon closer to reality, as sharing a complex and costly component becomes more beneficial due to drastic cost reduction. However it is still observed that, independent of the size and complexity, the crossover operator is likely to keep the modules in place until an alternative joint design leads to cost reductions that surpass commonality. Thus it can be concluded that for scenarios that modularity proves to be an effective strategy, the decomposition around the modules is dominated by the sharing decision and may be far from optimal.

When the modularity is a premise, the emergence of modules is usually a result of the inherent similarity of the products considered, and the success of the algorithms that are used to identify them. So the applications in Chapters 3 and 4 are to be carried out only if it is certain that it is worth implementing the modularity algorithms and complicating the process, as well as potentially getting far from the real optima due to

additional constraints. Introduction of cost models that quantify the benefit of modularity, however, clears this dilemma. It can be inferred that sharing of the modules is a dominant factor only before the mass production stage, that is, a point considerably down the cost curve (Figure 2.1), is reached. For products with relatively small production volume, sharing may lead to a drastic move along the curve, providing the necessary drive for modularity. When the algorithm is unable to find complex and costly components to share, the net cost reduction due to the commonality is small, which proves unable to force the populations toward modularity. So it is expected that the effect of modularity would be negligible for bigger production volumes, as the scale economies already provide a low average cost for each component. The modularity evaluation is not really needed for such cases. Of real interest is lower production volumes (scenario 1 and 2 in Chapter 5), and it turns out that modularity is an effective strategy for this situation, as proved with the ASF case study.

6.2 Contributions of the Thesis

- Even for simple products there is a vast number of possibilities to decompose the structures, and it is not feasible to make the decision based on past experience only. Using the novel systematic method developed in this study, not only the critical aspects of structural strength, assemblability and manufacturability are taken into account, but also the modularity strategy is integrated into the decomposition process.
- Identification of the modules is a combinatorial problem, and it is proved in this study that with a properly designed crossover operator, use of genetic algorithms is a very efficient way to search for a solution.
- This study focuses on solely structural products, and contributes significantly to the literature, as there are very limited reported examples on both joint design and modularity analysis applied to continuum-based or beam-based models of structures.

All of the milestones cited in Chapter 1 are reached at the end of this study, as summarized below:

1. Joint design is carried out efficiently using a joint library created by human experts, together with a representation and constraint set that guarantees only feasible decompositions are generated.
2. Important concerns of the designers are addressed for the early solution of assembly/manufacturing problems, by first integrating DFM and DFA heuristics and then a cost model into the criteria of the optimization problem.
3. Dynamics of the modularity strategy is fully understood. In the end of this thesis, benefit of component sharing is not taken for granted, but converted into a quantitative benefit, a reduction in production cost.
4. Developing the Pareto Set as an end result avoids the unpractical process of finding a single solution as a result of the optimization. It is rather in the interest of the designer and upper management alike to be able to visualize the trade-offs among the objective function terms as provided by this method.
5. The assembly synthesis method by the end of the thesis is easy-to-use and it requires no expertise on optimization. As no extensive human input is necessary and the genetic algorithm works efficiently enough, it is possible to generate quick but reliable results.

6.3 Future Work

- **Robustness:** Immediate future work will cover the area of *robust design*; note that the real application stage of the proposed method is conceptual design phase, when the design details are not very certain yet. It is therefore desirable to look for robust optimal solutions that are not very sensitive to dimensional changes; this is

especially critical for products with different overall sizes as in the case study given in Chapter 5. This problem was addressed by decreasing the tolerance for geometric similarity in ASF case study, but this may not be a solution for every application; especially when different parts of the structures are more prone to dimensional change compared to others. For the case when the cost models are integrated, robustness against changes in the production plans is loosely handled using different scenarios; more rigorous procedures should also be developed in this area.

- **More complex models:** Although the 3D beam models used in the case studies can provide useful insights to real automotive body designs, they are yet too simplified. Case studies with more detailed body models consisting of beams and plates would be desired to improve the applicability of the obtained results. It should be noted, however, that the current method can be applied to such beam-plate integral models with no modification, as long as product topology graphs are properly constructed and/or design variables and representation adequately convey the information about decomposition by specifying the right points divide the components into atomic members. A new joint library should also be appropriately designed for this case.
- **Expanded joint library:** While the joint and weld types cover basic joint variations found in many automotive bodies, the joint library is obviously not exhaustive. The inclusion of more types into the current joint "database" would be desired to enhance the applicability of the method. However the proliferation of joint types may make it necessary to present the cost of using each weld type, as in industry practice use of some joints may be deterring due to considerably increased set-up time, labor etc. If the cost models are at hand, this step certainly makes the method more realistic, as opposed to selecting weld types with respect to structural strength only.
- **Integration with assembly sequence planning:** A larger scale concurrent design tool can be developed by integrating the assembly synthesis with assembly sequence planning; though this requires additional iterative methods, probably an optimization

with multiple phases to choose the best assembly sequence given a decomposition. In case of limited computational resources or concern about reliability of two-stage optimization, a better idea may be incorporating guidelines, or if possible simple cost models into the assembly synthesis process that reflects some of the important criteria in the assembly sequence planning area.

- **Integration of other criteria:** It is obvious that the criteria used in this study do not fully address all of the important design criteria in structural design, specifically for automobile body structure design. Additional criteria such as stiffness and crashworthiness can be incorporated to increase the applicability of the method and the importance of the results to the designer. Due to different models and assumptions that this integration requires, it does not seem immediately practical to increase the number of objective function terms and address these problems. However once different tools to evaluate the decomposition with respect to other criteria are mature enough and capabilities as well as limitations are well understood, it will definitely be interesting for the design community to achieve a tool carrying out assembly synthesis with respect to all important structural criteria, as well as production cost.

APPENDICES

APPENDIX A

DETAILED RESULTS OF THE CASE STUDY IN CHAPTER 4

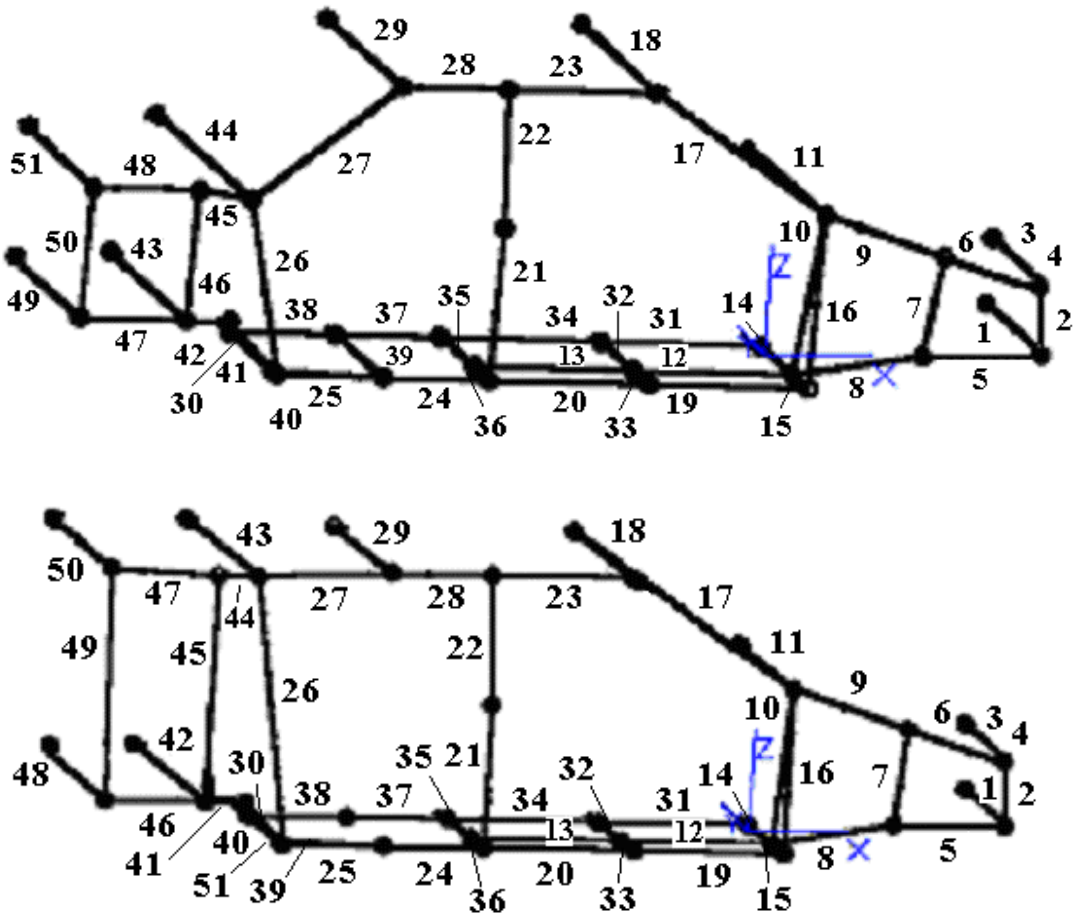


Figure A.1. Indices of beams for structures 1 and 2, in the case study of Chapter 4

Table A.1. Detailed decomposition results for global bending condition with lower weights in manufacturability (Figure 4.13).

Welds of Structure 1					Welds of Structure 2			
Beam 1	***Beam 2***	***Weld type***	***Projected Force [N]***		***Beam 1***	***Beam 2***	***Weld type***	***Projected Force [N]***
1	2	3	0		1	2	4	98.7
2	5	2	127.7		1	5	1	117.3
4	6	2	0		2	4	3	93.4
5	7	4	90.1		3	4	3	0
5	8	4	90.1		4	6	1	0
6	9	3	0		5	8	4	83.6
7	9	3	0		7	8	3	16.9
8	10	3	21.9		8	10	3	33.1
9	17	1	1152		9	10	4	0
10	12	4	16.7		9	16	3	0
10	17	3	0		9	17	4	30.8
12	13	3	0		10	12	3	0
14	31	3	0		10	17	3	0
15	16	3	145.5		12	13	3	0
15	19	4	1235		13	35	3	4070
16	17	1	756.2		13	36	4	718.8
17	23	3	159.5		14	31	3	0
18	23	4	191.3		15	16	2	809.9
19	33	3	724		15	19	3	0
20	33	4	0		16	17	4	236.7
20	21	3	0		17	23	3	146.3
20	24	3	0		18	23	3	0
21	24	4	0		19	20	4	0
25	26	4	392		19	33	3	718
26	52	4	0		20	21	3	0
27	28	2	1565.7		20	24	4	280.7
27	29	3	85.4		21	22	4	30
30	38	3	0		22	23	4	194.7
37	38	3	0		22	28	4	0
37	39	4	0		24	25	3	0
40	41	3	997.9		24	39	4	1400
45	46	4	0		25	26	4	0
45	48	3	129.2		26	51	3	587.5
48	51	4	0		26	44	4	957.1
50	51	4	0		27	28	3	0
					27	44	3	0
					28	29	4	0
					31	32	3	751.3
					31	34	4	0
					40	41	4	0
					44	45	2	2550.6
					44	47	3	139.3

Table A.2. Detailed decomposition results for global bending condition with higher weights in manufacturability (Figure 4.15).

Welds of Structure 1					Welds of Structure 2			
Beam 1	Beam 2	Weld type	Projected Force [N]		Beam 1	Beam 2	Weld type	Projected Force [N]
1	2	2	127.7		1	2	4	98.7
1	5	2	127.7		1	5	4	98.7
4	6	4	0.0		4	6	4	0.0
5	8	2	993.4		5	8	2	844.3
6	9	4	48.6		6	9	3	0.0
7	8	3	9.3		7	8	3	16.9
7	9	4	106.1		7	9	4	104.7
8	10	4	0.0		8	12	4	0.0
8	12	4	0.0		9	10	3	33.1
9	10	3	21.9		10	12	2	121.5
9	16	4	26.4		10	16	4	304.1
10	12	4	16.7		10	17	4	304.1
10	16	3	0.0		13	35	3	4070.0
10	17	4	221.7		13	36	3	0.0
12	13	4	519.3		14	31	3	0.0
13	35	4	0.0		15	16	4	0.0
13	36	4	723.9		15	19	2	1660.0
15	16	4	0.0		17	18	4	0.0
15	19	3	0.0		17	23	3	146.3
16	17	4	267.8		19	20	4	0.0
17	18	4	0.0		19	33	3	718.0
17	23	4	0.0		20	24	4	280.7
19	33	3	724.0		21	24	3	0.0
20	33	4	0.0		21	22	4	30.0
20	21	3	0.0		24	25	3	0.0
20	24	4	256.7		24	39	3	0.0
21	22	3	0.0		25	26	4	0.0
24	25	4	1843.0		26	51	4	0.0
24	40	3	0.0		26	44	4	957.1
25	26	3	0.0		27	28	3	0.0
26	52	3	398.0		27	29	4	0.0
27	28	4	284.1		27	44	3	0.0
28	29	3	13.7		30	38	2	0.0
30	38	3	0.0		31	34	4	0.0
31	32	4	0.0		32	34	4	0.0
31	34	4	0.0		37	38	2	0.0
37	38	4	0.0		40	41	4	0.0
37	39	3	0.0		41	45	3	1250.0
41	42	4	0.0		44	45	4	57.0
42	47	4	0.0		44	47	3	139.3
45	46	4	0.0		45	46	3	0.0
45	48	3	129.2		46	48	1	103.9
46	47	3	0.0		46	49	1	103.9

47	50	3	0.0		47	49	4	4.7
48	50	3	19.2		47	50	3	130.0
48	51	1	134.5					
49	50	1	19.2					

Table A.3. Detailed decomposition results for global torsion condition with lower weights in manufacturability (Figure 4.18).

Welds of Structure 1					**Welds of Structure 2**			
Beam 1	**Beam 2**	**Weld type**	**Projected Force [N]**		**Beam 1**	**Beam 2**	**Weld type**	**Projected Force [N]**
1	5	3	126		1	5	2	102.7
2	5	4	20.5		2	5	4	28.4
4	6	2	0		4	6	2	0
5	8	3	90.1		5	8	3	83.6
6	9	4	48.6		6	9	3	49.4
7	8	3	9.3		7	8	4	16.9
7	9	3	106.1		7	9	4	104.7
8	10	3	21.9		8	10	4	33.1
8	12	3	21.9		10	12	4	19.6
10	12	3	16.7		13	35	3	4070
13	35	3	3550		13	36	4	718.8
13	36	4	723.9		14	31	1	542.2
14	31	3	42.3		15	16	4	118.8
15	19	4	1235		15	19	4	1636
16	19	4	45.8		17	23	3	146.3
17	18	3	159.5		18	23	3	194.7
17	23	4	159.5		20	21	3	280.7
19	33	4	724		20	24	3	280.7
20	33	3	724		21	24	4	1292.7
20	21	3	256.7		21	22	4	30
20	24	3	256.7		22	23	4	194.7
21	24	3	810		22	28	3	292.5
22	23	3	191.3		25	26	4	178.6
22	28	3	309		26	51	4	587.5
24	25	4	1843		30	38	4	0
24	40	3	1843		34	35	2	4094.4
25	26	3	392		34	37	2	4094.4
26	52	3	398		37	38	1	0
27	29	4	85.4		40	41	3	0
28	29	4	13.7		41	46	3	1250
34	35	4	3550		44	45	4	57
34	37	4	3550		44	47	4	139.3
37	38	2	1856		45	46	4	3680.9
37	39	3	1856		46	48	3	53.4
41	42	3	0		46	49	1	103.9
45	46	3	63.4		47	50	4	130
45	48	3	129.2		49	50	4	51.1

47	49	2	45.8					
47	50	1	92.4					
48	50	3	19.2					
48	51	4	134.5					

Table A.4. Detailed decomposition results for global torsion condition with higher weights in manufacturability (Figure 4.20).

Welds of Structure 1					Welds of Structure 2			
Beam 1	Beam 2	Weld type	Projected Force [N]		Beam 1	Beam 2	Weld type	Projected Force [N]
1	2	2	310.7		1	2	1	125.6
1	5	2	310.7		2	5	3	83.0
4	6	2	0.0		4	6	2	0.0
5	7	4	76.5		5	7	3	1.7
7	8	2	1295.1		5	8	4	1.7
8	10	4	449.5		6	7	3	1.7
8	12	3	449.5		6	9	4	132.1
9	10	3	449.5		8	10	4	337.3
9	16	3	1382.4		8	12	3	337.3
10	12	4	8.4		9	10	3	337.3
10	17	4	161.4		10	12	4	27.8
13	35	3	203.0		10	16	4	99.6
13	36	4	232.4		10	17	4	99.6
14	31	2	1416.3		13	35	2	548.7
15	19	4	500.0		13	36	4	208.2
16	17	4	455.3		14	31	2	1169.9
16	19	3	749.2		15	16	4	728.8
17	18	4	209.4		15	19	3	429.5
17	23	4	209.4		17	23	4	543.7
19	33	4	232.3		18	23	1	390.0
20	33	3	232.3		19	33	4	208.8
20	21	3	317.3		20	33	2	503.1
20	24	4	317.3		20	21	1	498.5
21	24	4	1120.0		21	24	4	1261.0
22	23	4	66.6		22	23	4	61.2
22	28	4	909.3		22	28	3	581.1
25	26	3	926.1		24	25	4	1225.0
26	52	3	1484.0		24	39	4	1225.0
26	45	4	1159.6		25	51	4	214.1
27	28	4	902.7		26	51	4	2668.0
27	29	3	102.6		26	27	3	2226.1
27	45	3	102.6		26	44	2	2226.5
30	38	4	55.0		27	28	3	583.3
31	32	3	150.0		27	29	3	45.0
32	34	4	128.0		34	35	3	144.0
34	35	4	203.0		34	37	4	144.0
35	37	3	333.0		37	38	1	0.0

37	38	3	55.0		40	41	3	0.0
37	39	3	55.0		44	45	3	178.0
40	41	3	396.4		44	47	4	178.0
42	46	3	1316.0		46	48	3	198.0
42	47	4	1316.0		46	49	3	198.0
45	46	4	592.0		47	49	4	35.1
45	48	3	990.4		47	50	1	314.2
46	47	3	1051.0					

APPENDIX B

SUMMARY OF THE EMPIRICAL DATA ON WELDING OF ALUMINUM

Table B.1. Summary of weld and weldment properties of aluminum alloys.

Ref.	*Welding technique*	*Type*	*Aluminum alloy*	*Base metal tensile strength*	*Base metal elong.*	*Static failure load with weld*	*Elong. with weld*	*Fatigue failure load with weld*
[1]	Friction - stir	Lap	Alclad 2024-T3 Al7075-T6	475 MPa 595 MPa	- -	8-12 kN	- -	- -
[2]	MIG	Lap	5083 6082	335 MPa 322 MPa	15.3% 12%	- -	- -	10^5: 90 Mpa, 10^6: 40 MPa
[3]	Laser	Lap	5052 Al-4.5Mg	240 MPa 285 MPa	7% 30%	3.0 kN 3.5 kN	- -	- -
[4]	TIG	Lap	AlMgSi1	300 Mpa	-	-	-	10^5: 50 MPa, 10^6: 30 MPa
[3]	Laser	Butt	5052 Al-4.5Mg	240 MPa 285 MPa	7% 30%	208 MPa 279 MPa	4% 21%	- -
[5]	Laser	Butt	AlMgSi1	320 MPa	10%	230 MPa	2%	10^5: 60 Mpa, 10^6: 40 MPa
[6]	Laser	Butt	AlMg0.4Si1.2	270 MPa	-	-	-	10^5: 90 MPa, 10^6: 55 MPa
[6]	MIG	Butt	AlMg0.4Si1.2	270 MPa	-	-	-	10^5: 100 MPa, 10^6: 75 MPa
[7]	MIG	Butt	A5083P-O	314 MPa	24%	-	-	10^5: 70 MPa, 10^6: 35 MPa

References:

[1] Cederqvist and Reynolds, 2001.

[2] Ye and Moan, 2002.

[3] Matsumoto and Izuchi, 1995.

[4] Pinho da Cruz *et al.*, 2000.

[5] Behler *et al*., 1997.

[6] Matthes *et al.*, 1998.

[7] Ohta and Mawari, 1990.

APPENDIX C

DETAILED RESULTS OF THE CASE STUDY IN CHAPTER 5

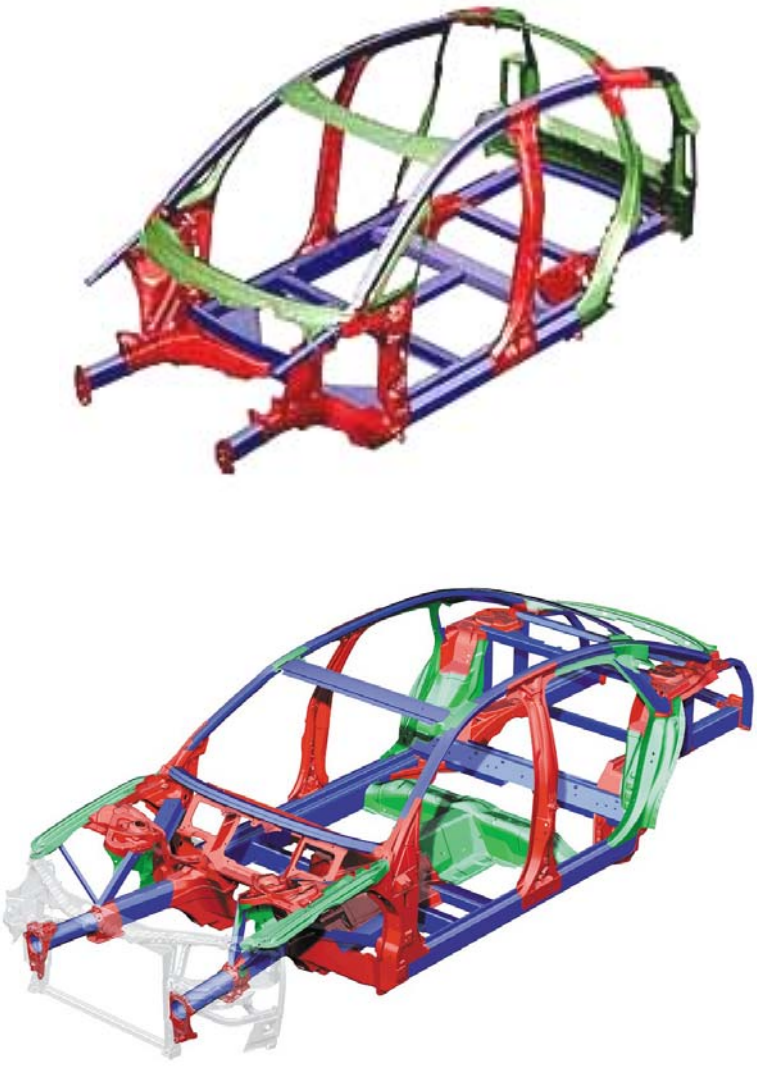


Figure C.1. Original Audi© A2 and A8 aluminum space frames.

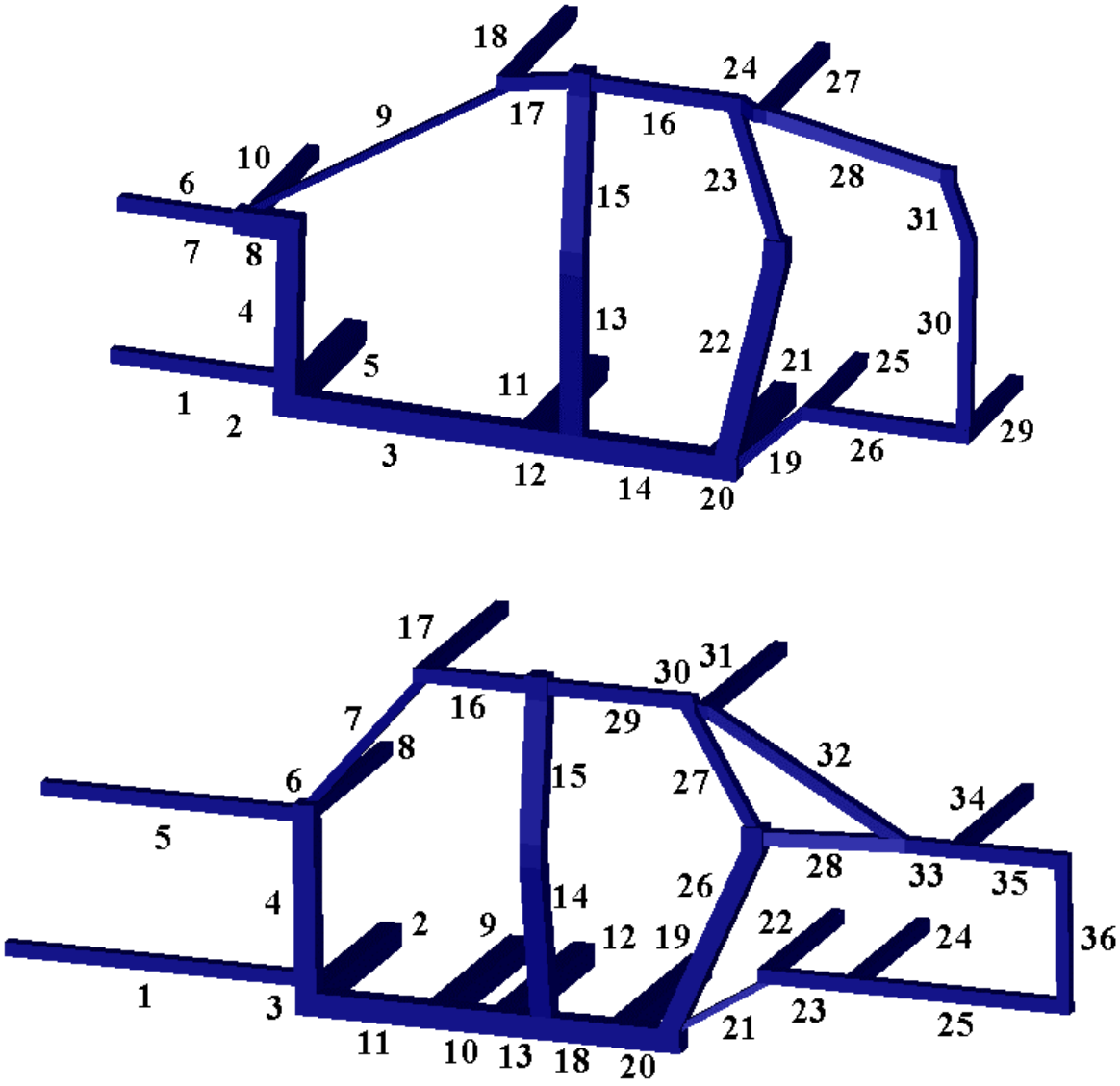


Figure C.2. Indices of beams for models based on Audi A2 and A8.

Table C.1. Cross sections of the beams for the models in Figure C.1.

Beams of Structure 1			Beams of Structure 2	
Beam index	Cross-section [mm*mm]		Beam index	Cross-section [mm*mm]
1	50*50		1	50*50
2	75*75		2	75*75
3	75*75		3	75*75
4	75*75		4	75*75
5	75*75		5	50*50
6	50*50		6	50*50
7	50*50		7	50*50
8	75*75		8	50*50
9	50*50		9	75*75
10	50*50		10	75*75
11	75*75		11	75*75
12	75*75		12	75*75
13	75*75		13	75*75
14	75*75		14	75*75
15	75*75		15	75*75
16	50*50		16	50*50
17	50*50		17	50*50
18	50*50		18	75*75
19	50*50		19	75*75
20	75*75		20	75*75
21	75*75		21	50*50
22	75*75		22	50*50
23	50*50		23	50*50
24	50*50		24	50*50
25	50*50		25	50*50
26	50*50		26	75*75
27	50*50		27	50*50
28	50*50		28	50*50
29	50*50		29	50*50
30	50*50		30	50*50
31	50*50		31	50*50
			32	50*50
			33	50*50
			34	50*50
			35	50*50
			36	50*50

Table C.2. Weld types and forces on welds, for point A in Scenario 1 (Figure 5.9).

Welds of Structure 1					Welds of Structure 2			
Beam 1	Beam 2	Weld type	Force on weld [N]		Beam 1	Beam 2	Weld type	Force on weld [N]
1	2	3	220.0		1	2	3	179.0
2	3	2	1164.9		3	11	4	1238.3
7	8	4	0.0		4	6	4	0.0
7	9	4	0.0		6	7	4	0.0
4	8	4	0.0		10	11	3	0.0
3	11	1	3114.6		10	12	1	2988.1
12	13	1	1343.1		13	18	3	167.0
15	16	1	1241.9		15	16	4	969.5
9	17	3	0.0		7	16	3	0.0
19	20	3	569.8		18	19	4	1265.1
14	21	4	1271.6		20	21	3	719.2
22	23	3	0.0		22	23	4	244.7
16	23	3	264.9		23	24	4	106.1
19	25	3	229.0		26	27	3	474.7
25	26	4	429.2		26	28	3	683.1
24	28	3	0.0		29	30	3	727.1
26	30	4	0.0		30	31	1	543.1
28	31	3	0.0		28	32	3	185.1
					33	34	4	106.1
					35	36	4	0.0

Table C.3. Weld types and forces on welds, for point B in Scenario 2 (Figure 5.10).

Welds of Structure 1					Welds of Structure 2			
Beam 1	**Beam 2**	**Weld type**	**Force on weld [N]**		**Beam 1**	**Beam 2**	**Weld type**	**Force on weld [N]**
1	2	3	220.0		1	2	3	179.0
2	3	2	1164.9		3	11	4	1238.3
7	8	4	0.0		4	6	4	0.0
7	9	4	0.0		6	7	4	0.0
4	8	4	0.0		10	11	3	0.0
3	11	1	3114.6		10	12	1	2988.1
12	14	3	278.0		13	14	3	1315.5
15	16	1	1241.9		15	16	4	969.5
9	17	3	0.0		7	16	3	0.0
19	20	3	569.8		18	19	4	1265.1
19	22	3	1619.5		20	21	3	719.2
14	21	4	1271.6		22	23	4	244.7
22	23	3	0.0		23	24	4	106.1
16	23	3	264.9		26	28	3	683.1
19	25	3	229.0		27	28	3	915.9
24	28	3	0.0		29	30	3	727.1
26	30	4	0.0		30	31	1	543.1
28	31	3	0.0		28	32	3	185.1
					33	35	3	0.0
					35	36	4	0.0

Table C.4. Weld types and forces on welds, for point C in Scenario 3 (Figure 5.11).

Welds of Structure 1					Welds of Structure 2			
Beam 1	Beam 2	Weld type	Force on weld [N]		Beam 1	Beam 2	Weld type	Force on weld [N]
1	2	3	220.0		1	2	3	179.0
2	3	2	1164.9		3	11	4	1238.3
7	8	4	0.0		4	6	4	0.0
7	9	4	0.0		6	7	4	0.0
3	11	1	3114.6		10	11	3	0.0
12	13	1	1343.1		10	12	1	2988.1
15	16	1	1241.9		13	18	3	167.0
9	17	3	0.0		15	16	4	969.5
19	20	3	569.8		7	16	3	0.0
19	22	3	1619.5		18	19	4	1265.1
14	21	4	1271.6		20	21	3	719.2
22	23	3	0.0		21	22	3	115.4
16	23	3	264.9		23	24	4	106.1
19	25	3	229.0		26	27	3	474.7
25	26	4	429.2		26	28	3	683.1
24	28	3	0.0		29	30	3	727.1
26	30	4	0.0		30	31	1	543.1
					28	32	3	185.1
					33	34	4	106.1

Table C.5. Weld types and forces on welds, for point D in Scenario 2 (Figure 5.12).

Welds of Structure 1					Welds of Structure 2			
Beam 1	**Beam 2**	**Weld type**	**Force on weld [N]**		**Beam 1**	**Beam 2**	**Weld type**	**Force on weld [N]**
1	2	3	220.0		1	2	3	179.0
2	3	2	1164.9		3	11	4	1238.3
7	8	4	0.0		4	6	4	0.0
7	9	4	0.0		6	7	4	0.0
4	8	4	0.0		10	11	3	0.0
3	11	1	3114.6		10	12	1	2988.1
12	14	3	278.0		13	18	3	167.0
15	16	1	1241.9		15	16	4	969.5
9	17	3	0.0		7	16	3	0.0
19	20	3	569.8		18	20	3	0.0
19	22	3	1619.5		20	21	3	719.2
14	21	4	1271.6		22	23	4	244.7
22	23	3	0.0		23	25	3	0.0
16	23	3	264.9		26	27	3	474.7
19	25	3	229.0		29	30	3	727.1
24	28	3	0.0		30	31	1	543.1
26	30	4	0.0		28	32	3	185.1
28	31	3	0.0		33	34	4	106.1
					33	35	3	0.0
					35	36	4	0.0

BIBLIOGRAPHY

Baldwin, D.F., Abell, T.E., Lui, M.-C. M., De Fazio, T.L. and Whitney, D.E, 1991, "An Integrated Computer Aid for Generating and Evaluating Assembly Sequences for Mechanical Products", *IEEE Transactions on Robotics and Automation*, Vol. 7, No. 1, pp.78-94.

Behler, K., Berkmanns, J., Ehrhardt, A. and Frohn, W., 1997, "Laser Beam Welding of Low Weight Materials and Structures", *Materials & Design,* Vol. 18, pp. 261-267.

Bendsoe, M. and Kikuchi, N., 1988, "Generating Optimal Topologies in Structural Design using a Homegenization Method", *Computer Methods in Applied Mechanics and Engineering*, Vol. 71, pp. 197-224.

Blackenfelt, M. and Stake, R.B., 1998, "Modularity in the Context of Product Structuring – A Survey", *Proceedings of the 2nd NordDesign Seminar*, 26-28 Aug, KTH, Stockholm, Sweden.

Boothroyd, G., Dewhurst, P. and Knight, W., 1994, *Product Design for Manufacture and Assembly*, Marcel Dekker, New York.

Cederqvist, L. and Reynolds, A.P., 2001, "Factors Affecting the Properties of Friction Stir Welded Aluminum Lap Joints", *The Welding Journal Research Supplement,* Vol. 80, No. 12, pp. 281-287.

Cetin, O. L. and Saitou, K. (a), "Decomposition-based Assembly Synthesis for Maximum Structural Strength and Modularity", *ASME Journal of Mechanical Design*. Accepted, to appear.

Cetin, O. L. and Saitou, K. (b), "Decomposition-based Assembly Synthesis for Structural Modularity", *ASME Journal of Mechanical Design*. Accepted, to appear.

Cetin, O. L. and Saitou, K., 2001, "Decomposition-based Assembly Synthesis for Maximum Structural Strength and Modularity", *Proceedings of the 2001 ASME Design*

Engineering Technical Conferences, Pittsburgh, Pennsylvania, DETC2001/DAC-21121, September 9-12.

Cetin, O. L., Saitou, K., Nishigaki, H., Nishiwaki, S., Amago, T. and Kikuchi, N., 2001, "Modular Structural Component Design Using the First Order Analysis and Decomposition-Based Assembly Synthesis", *Proceedings of the 2001 ASME International Mechanical Engineering Congress and Exposition*, New York, New York, November 11-16.

Cetin, O. L. and Saitou, K., 2003, "Decomposition-Based Assembly Synthesis of Multiple Structures for Minimum Production Cost", *Proceedings of IMECE'03, International Mechanical Engineering Congress and Exposition*, Washington DC, November 11-16. Accepted, to appear.

Chapman, C. D., Saitou, K. and Jakiela, M.J., 1994, "Genetic Algorithms as an Approach to Configuration and Topology Design", *ASME Journal of Mechanical Design*, Vol. 116, pp. 1005-1012.

Chickermane, H. and Gea, H.C., 1997, "Design of multi-component structural systems for optimal layout topology and joint locations", *Engineering with Computers*, Vol. 13, No. 4, pp. 235-243.

Chirehdast, M., Gea, H.-C., Kikuchi, N. and Papalambros, P.Y., 1994, "Structural Configuration Examples of an Integrated Optimal Design Process", *ASME Journal of Mechanical Design*, Vol. 116, No 4, pp. 997-1004.

Clark, J., 1998, "Future of Automotive Body Materials: Steel, Aluminum & Polymer Composites", Hoogovens Technology Day, Massachusetts Institute of Technology, Cambridge, MA. (http://msl1.mit.edu/hoog.pdf)

Coello, C.A.C., van Veldhuizen, D.A. and Lamont, G.B., 2002, *Evolutionary Algorithms for Solving Multi-objective Problems*, Kluwer Academic/Plenum Publishers, New York, 2002.

Constantine, B., 2001, "Economics of Tubular Hydroforming", Steering Committee Meeting Presentation, Materials Systems Laboratory, Massachusetts Institute of Technology, Cambridge, MA. (http://msl1.mit.edu/msl/meeting_04192001/prz_pdf/constantine.pdf)

Cross, A.D.J., Wilson, R.C. and Hancock, E.R., 1997, "Inexact Graph Matching using Genetic Search", *Pattern Recognition,* Vol. 30, No. 6, pp. 953-970.

Dahmus, J. B., Gonzalez-Zugasti, J.P. and Otto, K.N., 2001, "Modular product architecture", *Design Studies,* Vol. 22, pp. 409–424.

Davidson, J. A., "Design-Related Methodology to Determine the Fatigue Life and Related Failure Mode of Spot-Welded Sheet Steels," *SAE Technical Papers*, No. 8306-022, p. 539 – 551.

Davis, L., 1991, *Handbook of Genetic Algorithms*, Van Nostrand, Reinhold, New York.

De Fazio, T.L., Rhee, S.J. and Whitney, D.E., 1999, "Design-Specific Approach to Design for Assembly (DFA) for Complex Mechanical Assemblies", *IEEE Transactions on Robotics and Automation*, Vol. 15, No. 5, pp. 869-881.

Deb, K., Agrawal, S., Pratab, A. and Meyarivan, T., 2000, "A Fast Elitist Non-Dominated Sorting Genetic Algorithm for Multi-Objective Optimization: NSGA-II", *Proceedings of the Parallel Problem Solving from Nature VI Conference*, Paris, France, pp. 849-858. Springer, Lecture Notes in Computer Science No. 1917.

Eng, T.-H., Ling, Z.-K., Olson, W. and McLean, C., 1999, "Feature-based Assembly Modeling and Sequence Generation", *Computers & Industrial Engineering*, Vol. 36, No 1, pp. 17-33.

Fanjoy, D. W. and Crossley, W. A., 2002, "Topology design of planar cross-sections with a genetic algorithm: Part 1 – Overcoming the Obstacles", *Engineering Optimization*, Vol. 34, No. 1, pp. 1-22.

Fellini, R., Kokkolaras, M., Perez-Duarte, A. and Papalambros, P.Y., 2002, "Platform Selection under Performance Loss Constraints in Optimal Design of Product Families", *Proceedings of ASME DETC'02 Design Engineering Technical Conferences*, DETC2002/DAC-34099, Montreal, Canada, September 29 - October 2,.

Fisher, M., Ramdas, K. and Ulrich, K., 1999, "Component Sharing in the Management of Product Variety: A Study of Automotive Braking Systems". *Management Science,* Vol. 45, No. 3, pp. 297-315.

Fleury, C. and Braibant, V., 1986, "Structural Optimization: A New Dual Method Using Mixed Variables," *International Journal of Numerical Methods in Engineering*, Vol. 23, pp. 409-428.

Fujita, K. and Yoshida, H., 2001, "Product Variety Optimization: Simultaneous Optimization of Module Combination and Module Attributes", *Proceedings of the 2001 ASME Design Engineering Technical Conferences*, DETC01/DAC-21058, Pittsburgh, PA, September 9-12.

Garey, M. R. and Johnson, D. S., 1979, *Computers and Intractability, A Guide to the Theory of NP-completeness*, W. H. Freeman and Co., New York.

Globus, A., Lawton, J. and Wipke, T., 1999, "Automatic Molecular Design using Evolutionary Techniques", *Nanotechnology*, Vol. 10, No. 3, pp. 290-299.

Gupta, S., Das, D., Regli, W.C. and Nau, D. S., 1997, "Automated Manufacturability: A Survey". *Research in Engineering Design*, Vol. 9, No. 3, pp. 168-190.

Hahn, O., Gieske, D., Klasfauseweh, U., and Rohde, A., 1997, "Fatigue Resistance of Spot Welds under Multiaxial Loads," *Welding in the World*, Vol. 37, No. 5, pp. 15-22.

Hobbs, M.H.W. and Rodgers, P.J., 1998, "Representing Space: A Hybrid Genetic Algorithm for Aesthetic Graph Layout", FEA 1998 Frontiers in Evolutionary Algorithms,

Proceedings of the Fourth Joint Conference on Information Sciences, Research Triangle Park, NC, USA, Vol. 2, pp. 415-418, October 23-28.

Hopcroft J. E. and Wong J. K., 1974, "Linear Time Algorithm for Isomporphism of Planar Graphs," *Proceedings of the Sixth Annual ACM Symposium on Theory of Computing*, p. 172-184.

Huang, K.-I., 1993, "Development of an Assembly Planner using Decomposition Approach", *Proceedings - IEEE International Conference on Robotics and Automation*, Vol. 2, pp. 63-68.

Hyer, N.L. and Wemmerloev, U., 1989, "Group Technology in the US Manufacturing Industry: A Survey of Current Practices", *International Journal of Product Research*, Vol. 27, No. 8, pp.1287-1304.

Ishii, K., 1998, "The Life-cycle of an Enterprise", in A. Molina, J.M. Sanchez, and A. Kusiak (Editors), *Handbook of Life-cycle Engineering*, Kluwer Academic Publishers & Chapman and Hall, London ISBN: 0-412812509.

Jiang, T. and Chirehdast, M., 1997, "Systems approach to structural topology optimization: designing optimal connections", *ASME Journal of Mechanical Design*, Vol. 119, pp. 40-47.

Johanson, R., Kikuchi, N. and Papalambros, P., 1994, "Simultaneous topology and material microstructure design", *Advances in Structural Optimization,* Topping and Papadrakis (ed.), Civil Comp. Ltd., Edinburgh, Scotland, pp.143-149.

Kane, C. and Schoenauer, M., 1996, "Genetic Operators for Two-Dimensional Shape Optimization", *Lecture Notes in Computer Science,* Vol. 1063, pp. 355-369.

Kelkar, A., Roth, R. and Clark, J., 2001, "Automobile Bodies: Can Aluminum Be an Economical Alternative to Steel?", *Journal of The Minerals, Metals & Materials Society*, Vol. 53, No. 8, pp. 28-32.

Kim, K. and Chhajed, D., 2000, "Commonality in product design: Cost saving, valuation change and cannibalization". *European Journal of Operational Research*, Vol. 125, pp. 602-621.

Kota, S., Sethuraman, K. and Miller, R., 2000. "A Metric for Evaluating Design Commonality in Product Families", *ASME Journal of Mechanical Design,* Vol. 122, No. 4, pp. 403-410.

Lazzerini, B. and Marcelloni, F., 2000, "A Genetic Algorithm for Generating Optimal Assembly Plans", *Artificial Intelligence in Engineering*, Vol. 14, No 4, pp. 319-329.

LeBacq, C., Brechet, Y., Shercliff, H.R., Jeggy, T. and Salvo, L., 2002, "Selection of joining methods in mechanical design", *Materials and Design*, Vol. 23, pp. 405-416.

Li, Q., Steven, G.P. and Xie, Y.M, 2001, "Evolutionary structural optimization for connection topology design of multi-component systems", *Engineering Computations*), Vol. 18, No 3-4, pp. 460-479.

Malen, D.E. and Kikuchi, N., 2002, *Course notes from 'Fundamentals of Automotive Body Structures'*, Dollar Bill Copying, Ann Arbor, MI.

Mantripragada, R. and Whitney, D. E., 1998, "The Datum Flow Chain: A Systematic Approach to Assembly Design and Modeling", *Research in Engineering Design*, Vol. 10, No. 3, pp. 150-165.

Matsumoto, T. and Izuchi, S., 1995, "Laser Welding of Aluminum Alloy Sheets", *Kobe Steel Engineering Reports*, Vol. 45, pp.72-74.

Matthes, K.-J., Lubeck, K.-H. and Lanzendorfer, G., 1998, "Influence of Irregularities at Butt-Welded Seams on the Behaviour of the Vibration-Fatigue Strength of Sheet-Aluminium Joints", *Schweissen und Schneiden / Welding & Cutting*, Vol. 50, No. 3.

Messac, A., Martinez, M. P., and Simpson, T. W., 2002, "Introduction of a Product Family Penalty Function Using Physical Programming," *ASME Journal of Mechanical Design*, Vol. 124, No. 2, pp. 164-172.

Meyer, M.H., Tertzakian, P. and Utterback, J., 1997, "Metrics for Managing Product Development within a Product Family Context", *Management Science*, Vol 43, No. 1, pp. 88-111.

Muffatto, M. and Roveda, M., 2000, "Developing Product Platforms: Analysis of the Development Process". *Technovation*, Vol 20, pp. 617-630.

Nelson, S., Parkinson, M. B. and Papalambros P. Y., 2001, "Multicriteria Optimization in Product Platform Design", *ASME Journal of Mechanical Design*, Vol. 123, No 2, pp. 199-204.

Newcomb, P. J., Bras, B. A. and Rosen, D. W., 1998, "Implications of Modularity on Product Design for the Life Cycle", *ASME Journal of Mechanical Design*, Vol. 120, No. 3, pp. 483-490.

Nishigaki, H., Nishiwaki, S., Amago, T., and Kikuchi, N., 2000, "First Order Analysis for Automotive Body Structure Design", *Proceedings of the 2000 ASME Design Engineering Technical Conferences*, DETC2000/DAC-14533, Baltimore, Maryland, September 10-13.

Ohta, A. and Mawari, T., 1990, "Fatigue Strength of Butt Welded Al-Mg Aluminium Alloy: Tests with Maximum Stress at Yield Strength", *Fatigue and Fracture of Engineering Materials & Structures*, Vol. 13, No. 2, pp. 53-58.

Pinho da Cruz, J.A.M., Costa, J.D.M., Borrego, L.F.P. and Ferreira, J.A.M., 2000, "Fatigue Life Prediction in AlMgSi1 Lap Joint Weldments", *International Journal of Fatigue,* Vol. 22, pp. 601-610.

Radaj, D., 2000, "Fatigue Assesment of Spot Welds based on Local Stress Parameters," *Welding Research Supplement*, February, pp. 52 – 53.

Saitou, K. and Yetis, A., 2000, "Decomposition-Based Assembly Synthesis of Structural Products: Preliminary Results", *Proceedings of the Third International Symposium on Tools and Methods of Competitive Engineering*, April 18-21, Delft, The Netherlands.

Senin, N., Groppetti, R, and Wallace, D. R., 2000, "Concurrent Assembly Planning with Genetic Algorithms", *International Journal of Product Design and Process Development*, Vol. 16, No 1, pp. 65-72.

Shea, K. and Cagan, J., 1999, "Languages and Semantics of Grammatical Discrete Structures", *Artificial Intelligence for Engineering Design, Analysis and Manufacturing*, Vol. 13, pp. 241-251.

Sigmund, O., 2001, "A 99 Line Topology Optimization Code Written in Matlab". Structural and Multidisciplinary Optimization, Vol. 21, pp.120-127.

Simpson, T.W and D'Souza, B., 2002, "Assessing Variable Levels of Platform Commonality within a Product Family using a Multiobjective Genetic Algorithm", *Proceeding of the 9th AIAA/ISSMO Symposium on Multidisciplinary Analysis and Optimization,* AIAA 2002-5427, Atlanta, Georgia, September 4-6.

Simpson, T. W., Maier, J. R. A. and Mistree, F., 2001, "Product Platform Design: Method and Application". *Research in Engineering Design*, Vol. 13, pp. 2-22.

Skiena, S. S., 1998, *The Algorithm Design Manual*, TELOS/Springer-Verlag, New York.

Stake, R.B., 1999, "A Hierarchical Classification of the Reasons for Dividing Products into Modules - A Theoretical Analysis", *Licentiate thesis*, Department of Manufacturing Systems, Royal Institute of Technology, Sweden.

Sundgren, N., 1999, "Introducing Interface Management in New Product Family Development", *Journal of Product Innovation Management*, Vol. 16, pp. 40-51.

Suzuki, K. and Kikuchi, N., 1991, "A Homogenization Method for Shape and Topology Optimization," *Computer Methods in Applied Mechanics and Engineering*, Vol. 93, pp. 291-318.

van Holland, W. and Bronsvoort, W. F., 2000, "Assembly Features in Modeling and Planning", *Robotics and Computer-Integrated Manufacturing,* Vol. 16, No. 4, pp. 277-294.

van Vliet, J. W., van Luttervelt, C. A. and Kals, H. J. J., 1999, "State-of-the-art Report On Design for Manufacturing", *Proceedings of the 1999 ASME Design Engineering Technical Conferences*, DETC99/DFM-8970, September 12–15, Las Vegas, Nevada.

Wang, C.-H. and Bourne, D. A., 1997, "Design and Manufacturing of Sheet Metal Parts: Using Features to Aid Process Planning and Resolve Manufacturability Problems", *Robotics and Computer-Integrated Manufacturing,* Vol. 13, No. 3, pp. 281-294.

Whitney, D. E., Mantripragada, R., Adams, J. D. and Rhee, S. J., 1999, "Designing Assemblies", *Research in Engineering Design*, Vol. 11, No. 4, pp. 228-253.

Yao, Z., Bradley, H. D. and Maropoulos, P. G., 1998, "An Aggregate Weld Product Model for the Early Design Stages", *Artificial Intelligence for Engineering Design, Analysis and Manufacturing*, Vol. 12, No 5, pp. 447–461.

Ye, N. and Moan, T., 2002, "Fatigue and Static Behaviour of Aluminium Box-Stiffener Lap Joints", *International Journal of Fatigue,* Vol. 24, pp. 581-589.

Yetis, A. and Saitou, K., 2002, "Decomposition-Based Assembly Synthesis Based on Structural Considerations", *ASME Journal of Mechanical Design*, Vol. 124, pp. 593-601.

Yetis, A. and Saitou, K., 2000, "Decomposition-Based Assembly Synthesis based on Structural Considerations", *Proceedings of the 2000 ASME Design Engineering Technical Conferences*, Baltimore, Maryland, September 10-13, DETC2000/DAC-1428.

Yetis, A., 2000, "Decomposition-based Assembly Synthesis of Structural Products", *Master's thesis*, Department of Mechanical Engineering, University of Michigan-Ann Arbor.

Zugasti, J.P.G, Otto, K. N. and Baker, J.D., 2001, "Assessing Value in Platformed Product Family Design", *Research in Engineering Design*, Vol. 13, pp. 30-41.