MULTI-PLATFORM STRATEGY AND PRODUCT FAMILY DESIGN

Yanfeng Li

Dissertation submitted to the Faculty of the Virginia Polytechnic Institute and State University in partial fulfillment of the requirements for the degree of

Doctor of Philosophy

In

Industrial and Systems Engineering

Dr. Janis Terpenny (Chair)

Dr. Patrick Koelling

Dr. Asli Sahin-Sariisik

Dr. Subhash Sarin

February 22, 1010

Blacksburg, VA

Keywords: Product Design, Platform, Optimization, Costing

Multi-platform strategy and product family design

Yanfeng Li

Abstract

The application of product families and platforms has gained attention as a promising approach to achieving organizational objectives that provide customers with mass customized products while allowing for significant savings from commonality and reuse strategies. While the single-platform strategy has been widely studied, it may lead to the over expansion of the product family. Designers have to either continuously extend the exiting platform and/or impose strict constraints on new variants in order that there is a fit. On the one hand, continuously "extending" or "stretching" the platform forces the platform to become overburdened and less efficient. On the other hand, imposing strict constraints on new variants will force new variants to compromise performances.

In this research, the concept of a multi-platform strategy has been put forward to reduce or eliminate negative effects of the single-platform strategy by coordinating products in a complex product family into two or more platforms to provide enough product variety as well as commonality. The method is developed by adopting and synthesizing various tools and concepts from different research areas, such as design management tools, clustering analysis, statistics, decision analysis, mathematical programming, and engineering costing.

The product assets that can be shared by the products are determined through product asset value analysis and redesign effort analysis. The number of platforms is flexibly determined by a hierarchical clustering method based on product similarity/dissimilarity. The product-platform assignment problem is simultaneously solved during the clustering process. A multi-objective optimization model is formulated to determine the design specifications and address the product positioning. A Consistent Aggregate Function Formation Method (CAF^2M) is put forward to convert the multi-objective optimization model into a single-dimension problem that can quantitatively balance the tradeoff among the multiple objectives. To evaluate the economic benefit from the platform-based product development, an adjusted Activity-Based Costing approach is utilized to identify the cost savings with the consideration of learning effects.

A case application with seven automobile models is utilized to illustrate the proposed multi-platform strategy. The method was found helpful for determining and integrating critical design information into the design of product families and platforms.

Acknowledgement

Table of Contents

List of Figures

List of Tables

Chapter 1 : Introduction

1.1   Background

1.1.1 Present Situation in Product Family Design Research

Marketplace globalization, the proliferation of niche market, increased competitive environment, and needs for customized products have motivated industries from the design of individual products toward product family design. Accordingly, the production process has moved toward the mass customiza1tion process. In general, a product family is a group of related products that are derived from a product platform to satisfy a variety of market niches (Meyer & Lehnerd, 1997). Product platform is defined as "a set of common components, modules, or parts from which a stream of derivative products can be efficiently developed and launched" (Meyer & Lehnerd, 1997). Strategies of product family design have drawn attention from both academia and industries. Practices have shown that designing families of products takes advantage of economies of scale and scope as well as satisfies a variety of customer needs. Successful cases of designing product families can be found in various companies, including Black&Decker (Lehnerd, 1987), Sony (Sanderson and Uzumeri, 1997), and Volksvagan (Wilhelm, 1997).

Incorporating multiple products simultaneously distinguishes product family design from individual product design. When companies implement product family design, individual products and the coordination of the whole family are considered, the conflicts between product commonality/similarity and differentiation are addressed, and the product positioning in the market are coordinated as well.

Product family design requires additional care in planning over individual product design due to the external competition (with competitors' products in the market) and internal competition (with other products within the family). The platform structure focuses on balancing the commonality and differentiation across the product families as well as obtaining the optimal market positioning for individual products. Designing a product platform and its corresponding set of products is a challenging task that not only solves issues related to the product design, but also coordinates multiple products with the goal of increasing commonality across the family by extracting platform from these products without the loss of product distinctiveness.

The potential advantages and benefits of platform-based product development have been widely investigated. Four commonly known benefits are accepted by academia and industries. Firstly, product platforms can help companies introduce new products to market quickly and efficiently through component reusing and reconfiguring (Robertson and Ulrich 1998). Secondly, implementing product platforms can reduce product development time and system complexity, lower development and production costs, and improve the ability to upgrade products by sharing components and production processes. Thirdly, product platforms enable a variety of products derived easily and quickly to satisfy the needs and requirements from distinct market segments with fewer efforts (Pine, 1993). Finally, platforms can promote better learning across products and can reduce testing and certification of complex products, such as aircraft (Sabbagh, 1996), automobiles (Muffatto, 1999, Cusumano and Nobeoka, 1998, Bremmer, 1999, Bremmer 2000), spacecraft (Caffrey et al., 2002) and aircraft engines (Rothwell and Gardiner, 1990).

In the last decade, academic research has been seeking systematic methodologies to qualitatively and quantitatively address issues related to product platform and product family design, resulting in an active and continuously developing research area as shown in the survey paper by Simpson (2004). For example, Fujita and coauthors (1998, 1999, 2003, 2004) and Gonzalez-Zugasti and Otto (2000) focused on modular product family development with the consideration of module interfaces and Suh (2005) studied flexible platform design by providing adjustable product components. At the same time, industries are trying to offer opportunities to bridge the gap between theory and practice. For example, two important conferences that include academic and industrial participants were held, which provided a communication channel for academic and industry (Simpson et al. 2006). More detailed research can be found in the textbook edited by Simpson et al. (2005), which summarized the efforts of experts in academia and industry who are working to bridge the gap between (i) planning and managing families of products and (ii) designing and manufacturing them.

1.1.2 *Motivation*

To gain economies of scale and reduce the time-to-market, many manufacturers are developing similar products based on some common core technologies, which include forms, components, modules, interfaces and so on. Derived products from common core technologies are called product variants and shared core technologies compose the product platform, from which a stream of related and similar products is developed to fit different customers' needs. The platform-based product development is to design and develop various products that have something in common to meet needs in different market niches.

Currently, existing research is mainly focused on the research of a single-platform strategy (Figure 1.1), wherein the whole product family shares only one platform. There are only two options for the product-platform assignment: platform is either shared by all the products in the family or not at all. With the expansion of the product family, the drawbacks of the single-platform strategy have become apparent and brought fatal results to companies.

Figure 1.1: A Single-Platform Strategy
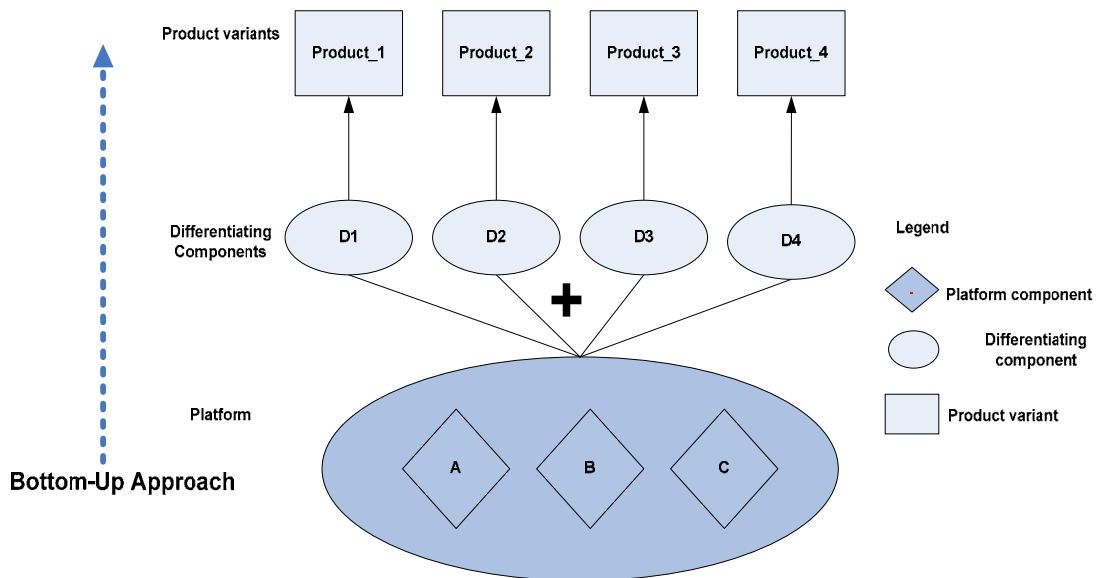
By sharing too many elements among different variants, variants are showing more similar features and not differentiating sufficiently from each other, losing their unique brand identity and further causing the loss of market competition. An unsuccessful platform example is the Chrysler K platform, from which a series of cars are derived. These cars were criticized by customers for lack of uniqueness (Carney, 2004). In addition, high-tech products are under-designed and low-tech products are over-designed by sharing across the product family, resulting in cannibalization of sales of high-end by

low-end variants (Cook, 1997). Thirdly, the product family is expanding and the platform is over-leveraged due to numerous products being assigned to the platform. Suh (2005) found that the number of product variants supported by a platform varied increasingly year after year and Weck (2005) claimed that the increasing trend was likely to continue in the future. Correspondingly, the platform has to accommodate a larger number of variants by continuously stretching the platform or sacrificing the individual performance to approach to the demanding requirements. The consequence is the loss of product attraction. What's more, the enlarging trend of the product family size makes the single platform strategy does not always satisfy the product requirements and bring positive consequences, especially when the gap of the required performance among products is very large (Suh 2005). When the product gap is too large, the compromising characteristic of platform strategy causes the failure of individual performance fulfillment.

Fortunately, manufacturers have realized that platforms cannot be stretched without restrictions. Also the accelerating mass customization and further partitioned market make the single-platform strategy not always practical and beneficial. The single-platform strategy has been proven beneficial in a certain environment. However, research has shown that it does not work quite well for a widespread product family. Under this situation, a multi-platform strategy (Figure 1.2) may overcome the shortages that single-platform strategy cannot solve.

Figure 1.2: A Multiple-platform Strategy
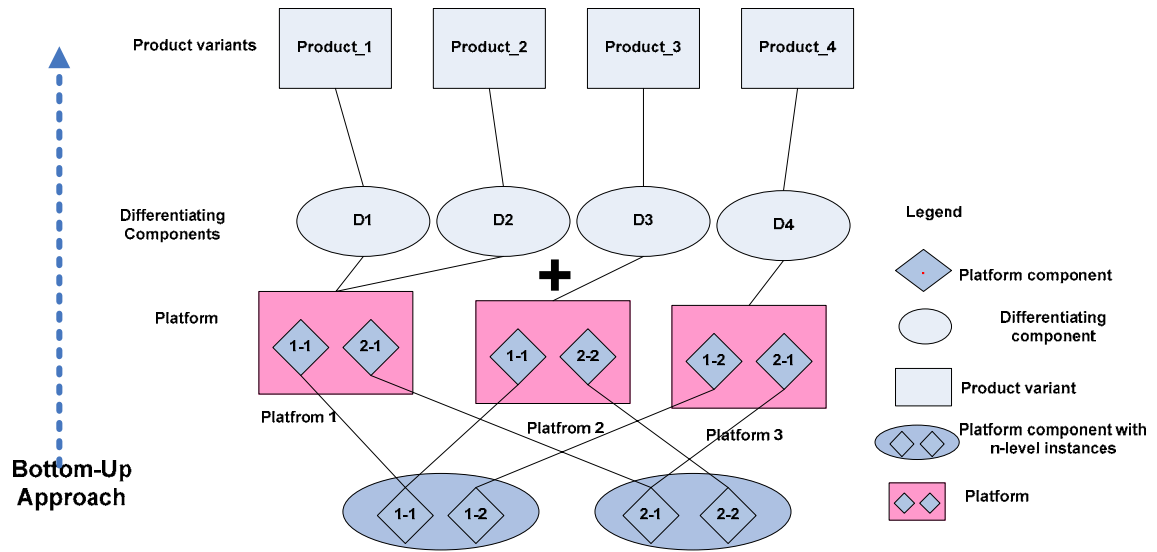
The concept of multiple-platform is that: the whole product family is supported by multiple platforms, and the product members are derived by picking appropriate platforms plus their distinct features; the components/parts/systems in the platforms are allowed to be same. In another word, the platform elements can be shared by a subset of product family (not necessarily the whole family) and individual products are derived from different platform element combinations in addition to their unique features.

In this research, the multi-platform strategy is investigated to simultaneously develop multiple products to satisfy requirements from various market niches. These products have their specific features, but they do share some commonality. The motivation of this research is to investigate the multi-platform strategy and explore methods to substantiate individual product variants from these multiple platforms.

6

1.2　Research Problems

It is apparent that the multi-platform strategy is more complex than the single-platform strategy. It should be considered from a strategic level. The product architecture, manufacturing, cost, engineering performance, values, demands and market competition issues should be taken into account.

The multi-platform strategy provides a method for product variants to share different levels of commonality of various features and components of the product family. Multiple platforms can also help to reduce the impact of commonality on performance of individual products.　Except the tradeoff between product commonality and distinctiveness, the following problems need to be addressed:

Problem 1:　Which parts/components/systems should serve as platform components in the multi-platform strategy?

Problem 2:　Given a set of product variants, what is the optimal number of platforms to derive them from?

Problem 3:　How to configure product members with platform elements to address the targeted market segments and competitors?

The first problem is a platform element determination problem, which is addressed by qualitatively and quantitatively analyzing product element values and redesign efforts. The second problem seeks to investigate the appropriate commonality level for the whole family as well as enough distinctness through technical and economic analysis. Once the identified platform elements and product commonality have been determined, the product-platform assignment problem is solved through a clustering analysis procedure

based on product similarity and product positioning problem is investigated by a mathematical optimization model.

In existing platform determination approaches, the platform setting is decided by quantitative or qualitative methods based on the analysis of common requirements for individual product, however, the roles, which individual products play in the whole family, such as added values and their contribution to the family, are not taken into account. According to Thevenot and Simpson (2004), the optimal commonality level can be obtained by minimizing non-value added variations across the products with a family without limiting the choices of the customers in each market segment. This research will explore a method that puts less value-added product elements into platform architecture.

Except questions 1-3, which are the primary issues at the heart of an effective multi-platform strategy, there are some other sub-problems to be solved.

- When addressing the three questions, how is the Voice of Customers (VoC) incorporated into consideration and how are the values of individual products expressed and deployed in the product platform planning?

- For designers or engineers, how are their preferences of the product components concisely expressed in the substantiation of platform and product variants?

- Given several platform alternatives, what metrics can be used to assess, identify and determine the most beneficial and promising one?

1.3 Research Purpose and Objective

The primary purpose of this research is to develop a generalized method that covers the determination of platform components, identifies the optimal platform setting for a large

product family under a multi-platform strategy, and configures the optimal product variants with the consistent preferences of design team. The secondary purpose is to provide a cost estimation model to evaluate the multi-platform strategy and investigate the cost saving as the result of implementing the multi-platform strategy.

1.4   Overview of Research Approaches and Methods

The research approach aids design teams in achieving three major tasks: 1) identifying the platform setting, given a set of existing products; 2) determining the values of design variables; 3) identifying the optimal platform setting.

In order to determine the platform elements and setting, a set of engineering decision management methods are applied to take the values of individual products and the values of product components into account. The platform components that can be shared by the products in a product family are determined according to their added values and redesign efforts. For the platform components, a data clustering approach is implemented to divide platform components in different product variants into smaller groups. The product variants in a same group will have the same platform component substantiation. Meanwhile, the products are assigned to the corresponding platforms.

To determine the values of design variables, a multi-objective optimization model with utility functions is formulated and a new method is proposed to form an aggregated objective function (AOF). A two-level consistency check technique in the new method ensures that individual and group decision makers will make consistent assessment during the AOF generation procedure.

To choose the most promising platform setting, multiple metrics, such as engineering cost and development, are used to evaluate the multi-platform strategy alternatives. Therein, an Activity-based Costing (ABC) approach is put forward to estimate the unit product cost and total manufacturing costs of the family. Since the same component can be mass produced and the manufacturing activities get faster with practice, the learning effects are considered to evaluate the potential cost saving due to the platform-based product development.

1.5 Potential Research Contributions

The contributions of this research will be to provide:

- A systematic method that can help to determine the common elements across the products by deploying product qualitative values into lower levels and analyzing product generational evolvement;

- A multi-objective optimization model with utility functions that can address various product requirements to facilitate the product positioning and express designers' preferences in specific market needs;

- A method that can handle multiple product attributes simultaneously and address their interdependencies with consistency check to convert a multi-dimension optimization problem into a single-dimension one;

- An adjusted Activity-based Cost estimation (ABC) technique presented to estimate cost savings by investigating available accounting system.

1.6   Organization of Document

The dissertation is organized as follows. Chapter 2 presents a review of literature, in which platform-based product design research is summarized from the customer domain to functional domains and further to design parameters domain. This helps explain how the research field has been exploited and why the research in this research is valuable and promising. Section 2.1 presents the literature review about product architecture and product planning. Section 2.2 discusses the design concept generation and evaluation tools, such as Quality Function Deployment (QFD) and Design Structure Matrix (DSM), and their applications in product family design. Section 2.3 presents the evaluation metrics that are used to assess the concepts of product family design as well as some well known decision making tools. Section 2.4 focuses on product family optimization models and the relevant literature is analyzed and categorized to briefly present their main contributions in exiting research. In addition, the application of multi-objective models in product family design is investigated. Section 2.5 concentrates on multi-platform strategies, which are organized by engineering-centric and management-centric views. Section 2.6 investigates cost estimation techniques, especially Activity-Based Costing (ABC) approaches, which are good at allocating and managing indirect costs across several products.

In Chapter 3, a brief introduction to the proposed methodology is presented along with detailed steps. Section 3.1 provides an overview of the proposed method and three phases of the method are illustrated step by step. Sections 3.2-3.4 present the proposed three-phase methodology. Specifically, Section 3.2 describes the platform identification

procedure, Section 3.3 illustrates the formation of a multi-objective optimization model and Section 3.4 provides assessment metrics for the multi-platform strategy.

In Chapter 4, an automobile product family is introduced to present the background of the case study. In the automobile product family, seven vehicles with great performance gaps are included and each of them covers a specific market segment. In this case study, Company X wants to find out whether the platform-based automobile development can satisfy the variety needs with least performance loss for individual products and at the same time, bring the benefits of economy of scale to the company.

Chapter 5 is a case study chapter, which presents how to implement the proposed method into the automobile product family. The seven automobile models, which target different customer groups with different performance needs, are studied, analyzed and optimized. The obtained results are analyzed and interpreted. The generated platform strategy is evaluated through the ABC analysis and the cost saving is assessed by the Net Present Value (NPV).

Chapter 6 is a closing chapter, which includes the conclusions that are reached in the multi-platform strategy. The research contributions and limitations are also addressed. In addition, the potential research opportunities in product family and platform design are presented.

Chapter 2 : Literature Review

In this chapter, issues related to the product family and product platform will be reviewed. Commonly used approaches and techniques, and their extensions will be discussed. In Section 2.1, product architecture and product platform planning are introduced to provide the foundation for the new method that will be put forward in Chapter 3. Design management tools and their applications in product family development are described in Section 2.2 to show their wide application in product design, especially in product family design. The selected design management tools include: Quality Function Deployment (QFD) and Design Structure Matrix (DSM). After that, literature about decision making tools design and assessment metrics for product family development are discussed in Section 2.3. Based on the assessment metrics described in Section 2.3, optimization models that determine the values of common and unique design variables are compared and classified in Section 2.4, especially the applications of multi-objective optimization models in product family design. To provide the background information about the multi-platform strategy, the existing research related to multi-platform strategies is presented and the existing models are categorized into management-centered and engineering–centered categories in Section 2.5. Section 2.6 investigates cost estimation techniques, which provide a basis for engineering-related cost estimation. Finally, Section 3.7 provides a brief summary of Chapter 2.

2.1   Product Architecture and Product Platform Planning

Product architecture is the structure that integrates components and subsystems of a product into a coherent mechanism to perform intended behavior and functions (Ulrich,

1995). It describes the ways in which functional elements of a product or system are assigned to its constituent sections or subsystems, and of the way(s) in which they interact. To realize a product, the overall functionality of the product is firstly decomposed into a set of defined functions and the component parts of the product that will provide those functions. Secondly, the interfaces between the components are specified (Ulrich, 1995). It also reflects rationale and intentions of the design such as functions, methods of use, methods of maintenance, and production. In other words, product architecture also has implications for how the product is designed, made, sold, used, repaired, etc. It has life-cycle influences on product development. Product development is a transformation from the needs of different market segments or customers to a product that is available to apply (Krishnan and Ulrich, 2001).

There is no doubt that product development is more complex and challenging in product family and platform planning (Robertson and Ulrich, 1998). Product family and platform planning needs to simultaneously balance various information, such as design inputs (market opportunity, customers' needs, etc), design outputs, and stakeholders' interests (Sahin, 2007). For example, product family design requires the trade-off between commonality and differentiation - the former will bring economical benefits to companies and the latter ensures products' attractiveness from different groups of customers (Robertson and Ulrich, 1998). Two architectures are available for product family and platform planning: integral and modular.

(1)   Integral platforms

As a scheme that allocates product functions to physical components, an integral architecture does not specify one-to-one mapping between functional elements and

components, which is the primary difference between integral architecture and modular architecture (Ulrich, 1995). It includes a complex mapping from functions to physical components and coupled interfaces. Integral architectures typically link subsystems with tightly coordinated relationships and distinctive or unique features that cannot be easily connected to other systems. Products with integral architecture tend to have complex and nonstandard interfaces, and the subsystems are built (or at least customized) explicitly for a particular product. For example, the distinctive identity of the Apple iPod music player is based on its integral architecture and it uses standard interfaces (such as the sound output for headphones or speakers), but the sound source is built into the unit and its internal components and software are relatively nonstandard (Fine, 2005).

In general, an integral platform consists of components that are shared by the product family and product variety is realized by adding unique components/features into the platform. Examples of integral platform are the telecommunication around network for spacecraft (Gonzalez-Zugasti and Otto, 2000) and automobile unit body (Ulrich, 1995). Since the physical components are coupled by the flows of material, energy, information etc. changes of one component in an integral platform will result in the adjustments of other related components.

(2)   Modular architecture

Ulrich and Eppinger (1995) defined a modular architecture as one that maps functional elements to the physical components of the products, and the interfaces between components are de-coupled. Baldwin and Clark (1997) provided a definition based on the relationship among components: A module is a unit whose structural elements are powerfully connected inside and relatively loosely connected to elements in other units.

So modules in a product are structurally independent of each other, but connected with each other. The concept of modularity helps break up a complex problem into small ones which are smaller and easier to deal with.

Fixson (2004) summarized product modularity from three views: systems, hierarchy and life cycle. The system's view considers the construction of modularity with respect to product development elements (components, functions, production processes etc.), and their interfaces. The hierarchical view illustrates the formation of product modules from top to down or from bottom to up. The life-cycle view studies modularity from the product life-cycle goals, such as product design and development, production, usage and retirement. Most of existing research related to modularity focuses on single product modularity, such as Dahmus et al. (2001), Strong et al. (2003) and Gershenson et al. (2004). 14 modular drivers (Appendix 1) are explored to deploy the design of modular products, and to facilitate the formation of modularity and life-time evaluation (Kreng and Lee, 2004).

In modular product family design, the product variety is realized by adding, substituting, and/or removing one or more functional modules from the platform. The concept of modularity is usually studied with product commonality or standardization. Fujita et al. (1998) recommended a customer's view, a function view and a manufacturing view and Tseng and Jiao (1998) suggested functional, behavioral and structural levels to explore and assess product modularity and commonality issues under the Product Family Architecture (PFA). Additionally, Thevenot and Simpson (2004) defined 19 metrics to evaluate modular product family (Appendix 2) and product platform, and some of them

are indirectly or directly derived from the modular drivers by Kreng and Lee (2004) and Otto and Holtta (2005), such as carryover, customization etc.

2.2 Design Management Tools

In this section, two commonly used design management tools are reviewed and their applications in product family design are discussed. Quality Function Deployment (QFD) and Design Structure Matrix (DSM) are presented respectively in Section 2.2.1 and 2.2.2.

Product design is a process which can achieve or realize products that meet or fulfill customers' functional and aesthetic requirements. Correspondingly, product family design is a process that can achieve a series of products simultaneously which satisfy a variety of requirements from different customer groups. In product family, the platform identification is the critical step to realize the product variants. How designers incorporate voice of customer (VoC) into platform formation has been addressed in existing publications. Critical Customer Requirements (CCR): a customer expectation regarding an aspect of a product or service (e.g., quality, speed, etc.). If the expectation is not met, the customers may be expected to refuse to purchase, or to purchase from a competitor. To determine the platform features, the CCR should be primarily satisfied. Customers' requirements are converted to design features by information representation and management tools, such as QFD and DSM that designers can clearly understand

2.2.1 Quality Function Deployment (QFD)

Quality Function Deployment (QFD) was originally developed by Yoji Akao in 1966 and provided a method to transform user demands into design quality, to deploy the functions forming quality, and to deploy a method for achieving the design quality into subsystems

and component parts, and ultimately to specific elements of the manufacturing process (Akao, 1994). House of Quality (HOQ) is used to pictorially depict the mappings.

Customers are considering about whether the product attributes meet their needs, while designer or engineers are thinking how to technically realize the products to satisfy customer requirements. Customers and designers express the same product in different terms and words and the QFD provides a team-based technique that identifies and translates customer requirements into technical specifications for product planning, design, process and production. It helps to map "what is desired" (Customer requirement) into "How to achieve" (technical specifications). By applying QFD with House of Quality, the customer requirements are transferred into engineering metrics, and the sequential engineering metrics are further mapped to the desired level. The hierarchy structure ensures that the customers' voice is expressed in a manner that the designers can manage (Ericsson and Erixon, 1999; Sahin, 2007).

To support the computational usage of QFD, the quantitative scale 1-5 or 1-9 are used to numerically represent the level of relationship (such as from very important to not important, from desired to undesired, from strong to weak) Quantitative values assigned to cells facilitate the numerical analysis about customer requirements and the strength of their relationships. To reflect the significance of customer requirements, functions and features, weights are assigned. For example, to facilitate the development of new product, Lin et al. (2006) proposed a procedure that comprises an Analytic Hierarchy Process (AHP), which is adapted to evaluate the importance of customer requirements, and an interpretive structural modeling (ISM) technique, which is used to tackle the interdependency of customer requirements so as to clarify their structural relationships.

Based on the real application, many extended QFD forms are developed, such as Function-behavior (FB) Matrix (Kumar and Allada, 2005), which is used to express the relationship between product functions and feasible behaviors (technologies), Modular Function Deployment (MFD) (Ericsson and Erixon, 1999), which is used to map the customer requirement to module level and Fuzzy QFD (Kim et al., 2000), which is use to describe the vagueness of customer requirements. In the next section, the QFD application in product design area will be investigated in more details.

2.2.1.1 QFD's Application in Product Design

The aim of the product design is to formulate a design that meets a number of customer requirements. Some researchers used quantitative QFD-based method to help designers improve customer satisfaction and obtain the optimal customer satisfaction levels. Fung et al. (2003) and Reich and Levy (2004) used mix-integer programming (MIP) and dynamic programming (DP) as well as QFD to solve multi-objective problem to get the optimal product quality under limited resources. Lai et al. (2006) developed a linear physical programming model with the help of QFD to maximize the overall customer satisfaction. Kreng and Lee (2004) have used QFD to deploy the design of modular products in two major phases. Phase 1 is the exploration of design requirements, which combines customer needs, company development strategies, and designers' preference to select proper modular drivers through competitive analysis. In phase 2, modular product analysis and linear integer programming are used to establish final module configuration.

Another QFD application is to capture design information as comprehensively as possible. It will help to catch changing information. For example, Marsot (2005) developed a QFD-based methodology to systematically integrate ergonomics at the

design stages. In addition, QFD has also been expanded to capture dynamically changing design information and relationships and to integrate design information relations and generate new design solutions. Adiano and Roth (1994) used looped QFD method to collect the updating customer satisfaction data and feedback the evolving requirements to the manufacturing process. To represent unclear customer requirements, Fung et al. (1999) incorporated fuzzy inference into QFD to accommodate the possible imprecision and vagueness in Voice of Customer (VoC) interpretation.

2.2.1.2 QFD in Product Family Design

In product family design, the applications of QFD are utilized with the identification of common customer requirements and the common customer requirements are cascaded into engineering-level features. The common customer requirements facilitate the information transmission and the identification of platform elements. For example, Zamirowski and Otto (1999), Kurtadikar et al. (2004), Zha et al. (2004), and Fung et al. (2007) utilized QFD to obtain platform elements by mapping the common requirements and functions to engineering characteristics. In addition, Zha et al. (2004), Xu et al. (2004), and Weck et al.(1997) imported fuzzy concepts into QFD to represent vague and imprecise information between customers and designers.

Martin and Ishii (2000, 2002) and Sahin (2007) applied QFD into multiple product design area. They use QFD and other techniques to identify the components which can serve as the platform components. By extending Kreng and Lee (2004)'s work to multiple-products design, Sahin (2007) used QFD to identify the critical common performance attributes for developing module-based platforms for a product family. In her research, QFD was used to identify critical product design assets for product platform and family

development. QFD was also used for collecting the changing market requirements. Fujita et al. (2003) used QFD to analyze the value distribution among a product family to define the appropriate value of respective products. Martin and Ishii (2000 &2002) use two-phase QFD to map product functions to components and calculate the index Generational Variety Index (GVI), a measure for the amount of redesign effort required for future designs of the products, and then standardized the components which are not likely to change in the future. These standardized components will serve as the platform components.

2.2.2 Design Structure Matrix (DSM)

Design Structure Matrix (DSM) is a matrix-based representation of a system or a project. DSM contains the constituent subsystems or activities and corresponding information exchange and dependency (Alizon et al., 2006). It is also referred to as Dependency Structure Matrix, Dependency Structure Method or Incidence Matrix.

According to the level of its application, there are four DSM applications useful to product developers, project planners, project managers, system engineers, and organization designers (Browning, 2001):

1)   Component-based DSM: used to represent the interaction among system components or product architecture;

2)   Team-based DSM: useful for organization structure relationship;

3)   Activity-based DSM: to model information flow among process activities;

4)   Parameter-based DSM: used to express the relationship among physical parameters.

DSM is a system analysis tool which can provide a simple, compact and visual representation of a complex system and capture the interactions or interdependencies between system elements. It is also a project management tool that helps to determine how the tasks in a project influence each other and helps managers to manage the project activities.

2.2.2.1 DSM in Product Design

DSM analysis provides insights into how to manage complex products. This tool captures the physical relationships between components in a product. It is usually used at the early stage of product design when the designers want to capture the interrelationships between design elements. The Design Structure Matrix (DSM) is a matrix where components are represented in rows and columns, where their interactions are expressed.

One of its applications in product design is to modularize products by identifying the connections among objects (Browning, 2001). If a group of components are physically connected to each other, then it is assumed that a module will be better than separate components. Pimmler and Eppinger (1994) built a DSM model which includes four types of relationship among product components: material flow, information flow, energy flow, and spatial flow. The product components with strong dependencies are clustered and modularized for reuse. The components are dependent within the modules and weakly dependent with other modules (Ericsson and Erixon, 1999).

2.2.2.2 DSM in Product Family Design

Sahin (2007) used DSM in product family development. She built up a functional DSM with four types of relationship proposed by Pimmler and Eppinger (1994) to define

design assets groups to form the common groups among a family of products. The relationship strength shows that the possibilities in which the components can be categorized in a module. In DSM, the cells on the diagonal have four types of relationships: Spatial (S), Material (M), Information (I) and Energy (E) and the cells in the non-diagonal cells show how strong the relationships are, corresponding to the four types of relationships. The strengths of each relationship are quantified following the 1-5 score defined by Pimmler and Eppinger (1994).

DSM analysis can also be used to manage the effects of change. For example, if the specification for a component has to be changed, it would be possible to quickly identify all processes or activities which are dependent on that specification, reducing the risk caused by out-of-date information. Yassine and Falkenburg (1999) proposed the Sensitivity-DSM (SDSM) to model how the changes can be propagated among the components of the systems. In the SDSM, matrix $A=\begin{bmatrix} A_1 & A_2 \\ A_3 & A_4 \end{bmatrix}$, and the elements in the cells are partial differentiation: $A_{ij} = \frac{\partial FR_i}{\partial DP_j}$, where A_{ij} means how functional requirements FR_i responds to the changes of design parameter DP_j, FR={[A], [DP]}, where A is design attributes. This Sensitivity DSM expands the traditional N-square DSM to $M \times N$ matrix.

In Kalligeros et al. (2006) adopted this sensitivity DSM to identify the non-sensitive components in order to form a robust platform for a family of products. Kalligeros et al. (2006) put forward a DSM-based method to qualitatively identify the platform on the component level. The DSM includes the external functional requirements (FRs) and factors affecting the design. The value in DSM represents how the FRs or design variables are affected by other FRs or factors. The idea behind is that the platform

variables are not directly sensitive to the changing FRs and insensitive to the customized (or differentiating) variables. The method can be applied at the system level, or components level. The sensitivity of the variable is presented by the partial derivatives of FRs and variables.

2.3 Platform Evaluation Tools and Metrics

2.3.1 Platform Evaluation Tools

The design of products or systems is a systematic process, which involves multiple techniques, tools, and skills. A best design comes from the comprehensive applications of knowledge, techniques, and tools. Engineering design involves decisions which alternative items, design principles or parts and the like are the best to use. Engineering is about making the best use of limited resources, so making a choice from a set of alternatives is necessary. Decision making techniques, both under certain and uncertain conditions, would provide substantial and strong support to engineering designers for a more subjective decision.

When the decisions are influenced by several conflicting criteria at the same time, such as cost criteria and performance criteria, decision making techniques based on multi-criteria are powerful to help out. These typical multi-criteria decision making tools include Analytic Hierarchy Process (AHP) (Satty, 1980), Utility Theory (Raiffa and Keeney, 1993) and Pugh's method (Otto and Wood, 2001). All the multi-criteria methods include the following procedure: defining the alternatives and decision criteria, evaluating alternatives upon each criterion, determining the preference to each criterion, obtaining

the overall scores for each alternative, performing sensitivity analysis and making a choice.

How to measure decision makers' preference is a great challenge in the multi-criteria decision making. Utility theory can be applied to show decision makers' favor to criteria during the decision making process. Utility theory, as a preference quantification technique, can facilitate the mathematical representation of design risks and designer preference, and get tradeoffs among conflicting design objectives. It is a powerful tool that can quantify the subjective preference into objective values. Utility is a measure that reflects satisfaction to one alternative relative to another. Three applications of utility theory (descriptive, normative and prescriptive) are described by Bell et al. (1988). Fernandez et al. (2005) studied the utility based decision support problem for multi-criteria design selection. In this paper, two types of attribute independency: utility independency (a designer's preference for levels of an attribute is constant) and additive independency (two attributes don't interact) are studied. A material selection problem is solved by the proposed method to validate its effectiveness. More Multi-Attribute Utility Theory (MAUT) methods could be found in Winterfelt and Edwards (1986) and Keeney and Raiffa (1993).

To include information vagueness and imprecision in the multi-criteria decision making, uncertainties and fuzzy concepts are incorporated. A decision tree is powerful tool that can obtain the expected monetary and non-monetary values of alternatives by studying alternatives under different scenarios and constructing a tree-structure to display the outcomes of each scenario (Sydenham, 2004). Weck et al.(1997) proposed an extended fuzzy AHP method to evaluate alternative production cycles. They also put a method to

check the consistency among the pairwise-comparison matrix. In addition, they defuzzified the analysis results by forming the surface center of gravity of any fuzzy set. Their method was applied in investigating three alternatives for a gear shaft case. Xu et al. (2004) used fuzzy theory to represent imprecise design information and designers' preference. They also constructed a fuzzy linguistic evaluation method to measure qualitative information. They used AHP to get weights on evaluation criteria and mapped into linguistic terms by scaling the importance measure to 0-1 range.

2.3.2   *Platform Evaluation Metrics*

Considering the integration of multiple design drivers, goals, constraints, and their relationships into design of any artifact including product platforms, it is impossible to evaluate design platforms and product families merely from a single perspective. Product performance criteria are widely used in product evaluation. Quality Function Deployment (QFD) is widely used to identify how the products satisfy customer requirements. Kreng and Lee (2004) have used QFD to deploy the design of modular products and explored 14 modular drivers (Appendix Table A.1) that facilitate the formation of product module from a life-cycle view.

Besides technical performance metrics, Martin and Ishii (2000) used Generation Variety Index (GVI) as a measure for modules to estimate the cost of changing platform module in platform to meet the stringent future engineering metric. Some literature presented metrics for measuring the success of platform and product family from the life-cycle view. Cagan and Vogel (2002) defined seven-class attributes: emotion, aesthetics, identity, ergonomics, impact, core technology, and quality. All of them address the product's usefulness, usability, and desirability. The ergonomics, core technology, and

quality attributes address the satisfaction of the product during use, both immediate and long-term. Social and environmental impact, product identity, and aesthetics each address lifestyle aspects of consumers. The emotion connects most directly with the consumers' fantasy in using the product.

In general, platform-based product family development provides a win-win approach for both customers and company with limited budget, development time, and manufacturing capacity and constraints. The intention of designers' is to create product solutions which can bring profits and at the same time, fulfill the variety of customer requirements and other standards (i.e., safety, government and environment regulations).

Thus a comprehensive and systematic approach to evaluate product platforms and product family derived from the platforms is necessary. Commonality is one metric that is widely used in platform determination and optimization problem. It is claimed by Thevenot and Simpson (2004) that the optimal commonality level is to reduce the non-value added variations as much as possible across the products with a family and at the same time to satisfy customers' choices to the fullest in each market segment. Thus, the commonality metric are used to measure the added-value of the platform modules. A variety of commonality index (See Appendix Table A.2) have been developed and comprehensively reviewed by Thevenot and Simpson (2004). Using commonality as a metric to assess platform strategy is based on the assumption that the more commonality in the family, the more economic profit it will bring to the company. This assumption is not always true. The higher commonality level, the more performance loss will happen. Consequently, the products have less attraction to customers, causing the loss of market.

In addition, there is no consensus on what level of commonality is the best. Thus the commonality level is not always a good or appropriate metric.

Ye et al. (2005) and Otto and Holtta (2005) both evaluated product family development from the life-cycle perspective. Ye et al (2005) presented ten factors that affect product family design. These factors include customization, market life, technological innovation, family size, complexity, development time, service, and maintenance, environmental impacts, manufacturing cost, and product volume. Similarly, Otto and Holtta (2005) proposed a framework of 19 metrics for multi-criteria product platform evaluation, which combines the metrics from modularity, platform and general product development. The metrics are grouped into six categories: customer, variety, flexibility, complexity, organization, and after-sale. The metrics in Otto and Holtta (2005) can also be used as the measures to evaluate product modularity since modularity is closely connected to products and modular products are more applicable to implement platform strategy. The life-cycle evaluation ensures a comprehensive analysis and makes a reliable choice.

2.4   Product Family and Platform Optimization

Optimization methods have been widely used for decades during product design phase to help determine the values of the design variables and to minimize (or maximize) the objective values while satisfying a set of constraints (technical or non-technical constraints). When optimizing a product family, a set of products are optimized simultaneously instead of individual product optimization. At this time, the problem formulation is expanded to find out the values of common design variables (platform variables) and unique variables for each product (differentiating variables). Product

family optimization becomes much more complex than individual product optimization. In product family optimization, the designers have to balance the commonality (from the manufacturer's view) and the performance (from the customer's view). The problem is formulated with a variety of engineering-related objectives, such as, reducing cost and simplifying the design effort (Simpson, 2004) improving life-cycle design (Ortega et al. 1999), optimizing production cost or profit as well as reducing development time to market (Krishnan and Ulrich 2001), optimizing product portfolio planning and positioning (Jiao and Zhang, 2005) and maximizing utility benefit per cost/profit (Yano and Dobson, 1998).

Product family development will bring the economy of scale to companies and cost factor is one of principal issue to consider. Engineering-related cost minimization is the common objective in the product family optimization problems. Choubey (2007) use Genetic Algorithm (GA) techniques to solve the single- and multi-platform questions by minimizing the total production cost. In his thesis, he studied the product components and realized product variants by adding or removing components. The platform components are known as priori. Also the product demand uncertainties are studied by considering different uncertain scenarios.

Fujita and his coauthors studied product family and platform under modular architecture. Fujita et al. (1999) studied the product family development under modular architecture and identified common modules by simulated annealing method. They assumed the product functional modules with feasible levels are predefined and the objective is to find the optimal module combination to realize product variants as well as to minimize the engineering cost. Three basic constraints are introduced to represent the relationship

between modules: diversion feasibility constraint, simultaneity constraint and capacity constraint. Fujita et al. (2001, 2004) investigated a simultaneous product variety optimization method, which combined GA, MIP with Branch-and-Bound technique, and nonlinear programming to find out the common module and module attributes. An airplane example is used to demonstrate its validity and effectiveness.

Fellini et al. (2004) proposed a method to identify the product platform and optimize the family products, based on the variable sensitivity from individual optima. Their method basically can be divided into: 1) generating variants from requirements and optimizing individual products; 2) identifying platform variables based on design variables' sensitivity analysis and optimize the variants; 3) evaluating design alternatives and product family. Their method is based on the assumption that the robust product family design is realized by sharing the most insensitive design variables with less performance loss for product variants.

Depending on whether the designers need to determine the common components, two kinds of optimization methods are put forward: platform determination problems (such as Fujita et al., 1999) and individual platform instantiation problems (Simpson et al. 2001; Nelson et al. 2001; Rai and Allada, 2003). In the first kind of problems, the designers need to answer the following questions: what components should be included in the platform? What are the optimal values of the common design variables in the components? The design challenge is to determine (select) the platform that will generate family design with minimum deviation from individual optima. The tradeoff between maximizing commonality and minimizing individual performance deviations should be balanced. The second kind of problem is optimizing the individual products based on the

platform. This kind of problems often come with the first kind of problems: platform setting should be given as a priori and the values of design variables are left to specify. Thus product family optimization can be classified as priori or a posteriori optimization. A "priori" optimization means the design variables that define the product platform are known before performing the optimization, while a "posteriori" optimization means which design values in the platform are determined during the optimization procedure.

Posterior optimization models are widely formulated by researchers, such as Simpson et al. (2001), Nelson et al. (2001), Rai and Allada (2003) and D'Souza and Simpson (2003). Therefore the focus of the review about optimization is the "posteriori" optimization. They would like to determine the platform settings and their values in the optimization procedure.

Accordingly, two alternative approaches for optimizing the product platform and product family members are developed: first identifying the platform settings and then instantiating the individual products by specifying design variables (two-stage approach) or addressing platform settings and optimizing design variables simultaneously (single-stage approach). Single-stage approach can optimally solve the problem related to product platform and product family simultaneously and can include the combinational selection of platform variables and the determination of the values of the individual products, while two-stage approaches complete the two tasks separately.

Various solution methods for the platform optimization problem were implemented, including (but not limited to) Branch and Bound algorithm (Fujita and Yoshida 2001), Dynamic Programming (Allada and Jiang 2002), agent based techniques (Rai and Allada 2003), Simulated Annealing (Fujita et al. 1999), Genetic Algorithms (Fujita and Yoshida

2001; Li and Azaram, 2002, Simpson and D'Souza 2002, Simpson and D'Souza 2004, Jiao and Zhang 2005).

2.4.1 One-stage Approaches

One-stage approach is widely used in scale-based platform product family design optimization. Messac et al. (2002b) proposed a single-stage approach to model a multi-objective optimization problem. In their paper, the physical programming method was used to solve the problem. D'Souza and Simpson (2003), Simpson and D'Souza (2004) and Akundi et al. (2005) used Genetic Algorithms (GA) to balance the commonality in the family and desired performance of individual products. The one stage approach is a method solving the mixed problem with combinational problems and individual product optimization problems. Though the resulted solutions are relatively better than those obtained by two-stage approaches (Simpson, 2005), the computational complexity is higher than the two-stage approach and a powerful method needs to be developed to reach the optimality efficiently.

Fujita involved a series of module-based product family design research. Fujita et al. (2004) used one-stage approach in modular platform design and product family design optimization. Fujita (2000) classified modular product variety problem into three classes: optimizing module attributes under fixed module combination, optimizing module combination under predefined module candidates, and simultaneously optimizing both module attributes and module combinations. Third class questions are optimized by Fujita et al. (2004). In their model, three sub-models are involved: commonality and similarity pattern, similarity direction and module attributes optimization model. Genetic algorithm (GA), Branch and Bound technique (BB), and Successive Quadratic

Programming (SQP) were used to in the three sub-models respectively. A set of tentative solutions that are generated in commonality and similarity model were evaluated by the BB and SQP techniques.

2.4.2 Two-stage Approaches

The two-stage optimization approach can be found in Messac et al. (2002a). The design variables which most largely impact the product performances were identified and make them as the unique variables at the first stage and a physical programming method was used to find the optimal values of both common variables and unique variables at the second stage.

Nayak et al. (2002) proposed a variation based method to minimize the deviation of design variables while satisfying the range of performance requirements. In the first stage, the design variables with the acceptable standard deviation/mean ratios were picked as the common design variables and were used to initiate the individual products in the second stage.

Dai and Scott (2007) applied sensitivity analysis and cluster analysis method to determine the platform settings first and then design the whole family. In their method, the products are optimized individual, then the sensitivity analysis was performed for each product to identify the less sensitive design variables and cluster analysis method is used to determine the common variables with regard to the performance loss due to the commonalization. After that, the individual products are optimized based on the determined common variables.

Fujita et al. (1998) put forward a module-based two-stage optimization approach, in which common modules were identified through product requirement analysis and the module attributes were optimized for individual products sequentially.

Based on product architecture,, the platform-based product family planning can be divided into module-based product family (product varieties are instantiated by adding, substituting, and/or removing one or more functional modules from the platform) and Scale-based product family (product varieties are realized by stretching or shrinking the platform in one or more design variables to satisfy various requirements). The literature related to product family optimization is organized by one-stage and two-stage approaches with consideration of product architecture. The approaches in the existing literature for product family design optimization problems are summarized in Table 2.1. From Table 2.1, one-stage platform optimization approaches are most used and scale-based platform is widely studied by researchers.

2.4.3 Multi-objective Optimization (MOP)

In the product family design, a set of products are targeted at the same time. For each product, the designers want to get the optimal performance output for the aiming market niche subject to the several technical constraints. When the product family is designed, there are often a number of design objectives to be considered. These objectives are sometimes conflicting and no design can be considered as the best with respect to all objectives in each product. These considerations have led to the application of the multi-objective optimization techniques in product family design.

Table 2.1: One-stage and two-stage approaches classification

	Scale-based platform	Module-based platform
One-stage approach	Messac et al. (2002b); D'Souza and Simpson (2003); Simpson and D'Souza (2004); Akundi et al. (2005); Simpson (2005)	Fujita et al. (2004),
Two-stage approach	Nayak et al. (2002); Dai Z. and Scott (2007) ;	Fujita et al. (1998), Fujita and Yashida (2001)

Multi-objective optimization (or programming) (Sawaragi et al., 1985), also named as multi-criteria or multi-attribute optimization, is defined as the process of simultaneously optimizing more than one conflicting objectives subject to a set of constraints. Multi-objective formulations are realistic models for many complex engineering optimization problems. In many real-life problems, objectives under consideration are conflicting with each other, and optimizing a particular solution with respect to a single objective can result in unacceptable results with respect to other objectives.

Without loss of generality, all objective functions in MOP can be formulated into minimization type, and a minimization type multi-objection decision problem with k objectives is defined as:

$$Min\, Z(X) = \{Z_1(X), \dots, Z_k(X)\}$$

$$Subject\, to:$$

$$G(X) \leq 0;$$

$$H(X) = 0;$$

$$X_l \leq X \leq X_u$$

where $Z_i(X)$ is the ith objective, X is n-dimensional decision variable vector $X=(x_1, x_2, \dots, x_n)$, $G(X)$ and $H(X)$ are a set of inequality and equality constraints

respectively and the last constraint represents the lower bound and upper bound of the decision variables.

In multi-objective optimization problems, there should be no single solution that can simultaneously make all the objectives maximize or minimize the objective to their fullest. But there exist a set of equally efficient or non-inferior, alternative solutions, known as the Pareto Optimality (Liu et al., 2002). Nelson II et al. (2001) studied the application of multi-criteria optimization problem in product platform identification and discussed the Pareto sets that corresponded to various product derivatives to a systematic methodology for design decision making. For a minimization problem, a feasible point p* is Pareto optimal if and only if there exists no other feasible point such that :

$$(1)\ z_i(p) \leq z_i(p*) for\ all\ i = 1,2,\dots,k$$

$$(2)\ z_i(p) < z_j(p*) for\ at\ least\ one\ j$$

From the definition, the Pareto optimality means that for the point in Pareto optimal set, one objective value cannot be improved without scarifying at least one of the rest objectives.

There are two general approaches to multiple-objective optimization. One is to combine the individual objective functions into an aggregated objective function or make all but one objective to the constraint set. In the former case, determination of a single objective is possible with methods such as additive/multiplicative utility functions, weighted sum functions, etc., but the problem lies in the proper selection of the weights or utility functions to characterize the decision maker's preferences. In practice, it can be very difficult to precisely and accurately select these weights, even for someone familiar with

the problem domain. In addition, small changes in the weights can sometimes lead to quite different solutions. Seepersad et al. (2000) formulated the product family problem into a decision support problem, a specific multi-objective optimization problem which tries to minimize the total performance deviation from the targets values with predefined weights. Seepersad et al. (2002) used utility function to combine the multiple objectives together to form an aggregated objective. Messac et al. (2002) proposed a single-stage approach using physical programming to trade-off the individual performance loss and commonality requirements. In the second type of approaches, the multi-objective problem is formulated to move all objectives except one to the constraint set, and a constraining value must be predefined for each of these former objectives. Which objective(s) will be moved to the constraint set is arbitrarily selected. In addition, these approaches need the decision makers to specify the acceptable levels for the "constrained" objectives.

In general, the multi-objective optimization problems are converted into single-objective problems, which have been well studied. An Aggregated Objective Function (AOF) is one function that combines multiple objective functions and decision makers' preference into one function. It is regarded as the objective function to represent the original objective set. Just as Dai and Scott (2006) mentioned, using AOF was more suitable for the relatively small product family and could reduce the computational expense. However, most of AOFs are formulated by pre-determining linear weights for the individual objective functions and a single solution obtained from one AOF could not exactly reflect all trade-offs, especially under the condition where the preferences vary with the outputs of individual objective values.

The second general approach is to find out an entire Pareto optimal solution set or a representative subset. A Pareto optimal set is a set of solutions that are non-dominated with respect to each other. While moving from one Pareto solution to another, there is always a certain amount of sacrifice in one objective(s) to achieve a certain amount of gain in the other(s), which means the improvement of the values of one objective functions cannot be obtained without the degradation of the other(s). Pareto optimal solution sets are often preferred to single solutions because they can be practical when considering real-life problems, but great computational efforts are required.

Nelson et al. (2001) constructed a multi-criteria optimization model, which was demonstrated in the application of the nail gun family design, to balance the trade-off between platform commonality and the performance of individual platform, and investigated the Pareto optimal set for each possible combination of sharing parts. Dai and Scott (2006) presented a multi-objective model that obtained the values of common design variables and non-platform design variables in two separate stages and a preference aggregation approach was used to obtain the Pareto optimal set by finding the indifferent points among objective functions. Pareto optimal sets can be of varied sizes, but the size of the Pareto set usually increases with the increase in the number of objectives.

Although multi-objective optimization models and their extensions have been used in product platform design problems, they are limited in the single-platform strategy. For the product family with a large number of product variants, the number of objectives is large and thus the size of the Pareto optimal set would be correspondingly large and great computational efforts are needed. In addition, the Pareto sets are not always necessary.

Single-objective optimization models have been broadly studied and a bunch of techniques and algorithms can be applied. In addition, various software packages are available. Thus converting multi-objective problems into single-objective problems would be an applicable way to solve the tradeoffs among objective functions. In this research, a new approach is put forward to generate AOF from the conflicting multiple objectives. The approach also ensures obtaining consistent preferences from decision making groups.

2.5 Multi-platform Strategy

The multiple platform optimization problems are treated from either management-centric or engineering-centric view. It is apparent that engineering rarely has access to and expertise in market-related topics. Conversely, management does not have a detailed understanding of the technical issues resulting from various platforming decisions. In this section, the multiple-platform problem review is organized from these two engineering and management aspects. Martin and Ishii (2000&2002), Kumar and Allada (2005) and Sahin (2007) did some research in the first class (management-centric) and work by Seepersad et al. (2000&2002), Dai and Scott (2007) and Simpson (2005) belongs to engineering-centric.

2.5.1 Management-centric Multiple-platform Identification Approaches

Martin and Ishii (2000&2002) put forward an index-based platform design method to meet the changing market environment. In their method, the platform design was originated by market changes for a component such as the changing customer requirements, regulations and competitor introductions etc. Spatial index CI, which

represented component correlations, and generational index GVI, which captured evolving information, were used as platform identification criteria to determine platform components. Based on two indices, illustrative examples of an inkjet printer (Martin and Ishii, 2000) and water cooler (Martin and Ishii, 2002) are analyzed with the consideration of design related costs.

In their method (Martin and Ishii, 2000&2002), different component combinations are analyzed based on GVI and CI indices, and multiple platforms can be developed corresponding to each of the potential standardized components. However, they didn't provide a clear criterion on when multiple platforms are needed, how many platforms would be most beneficial and how to assign current and future product variants into platforms. What's more, though design and redesign costs are taken into account during the platform design process, they don't quantitatively include the product demands and competition in the platform design procedure.

Kumar and Allada (2005) formed multiple platforms through analyzing product functions and feasible behavior (or technologies) with the goal of balancing the tradeoff between the customer satisfaction (from customer view) and switchover cost (from firm views) simultaneously based on interpreted customer needs over a certain time period. They assumed the product demand data is known as a priori and there is no market competition incorporation. The FB-ACO method by Kumar and Allada (2005) is capable of determining the best possible number of platforms and their respective configurations. The number of platforms is determined in the optimization procedure. In addition, they assume the product demand data is known as a priori and there is no market competition incorporation.

Sahin (2007) presented a modular-based multiple platform formation method. Originated from customer requirements and critical assets group, modular product architecture as well as platform modules are identified with the help of QFD, DSM and K-means clustering analysis. For each assets group, multiple modules are formed through product similarity analysis. Module Indication Matrix (MIM) is used to determine the platformability of the modules. Also she presented a comprehensive analysis about the influence and significance of customer needs, and market segment on the platform determination procedure. The platform configuration is decided by the combination of the generated modules, based on the collected comprehensive information and the platform determination problem, and platform-to-variant assignments are addressed. Similarity of the product attributes determined the number of platforms. Her research makes a good start for future numerical multi-platform research, though she did not qualitatively address issues of the market demand, competition, and business-related by providing the cost values and product values.

2.5.2 *Engineering-centric Multiple-platform Identification Approaches*

Seepersad, Hernandez and Allen (2000) presented a quantitative method to determine the number of scalable platforms for a specific market. A decision support problem (a problem that tries to minimize the weighted deviation from the target values) was formulated with predefined platform settings for various production capacity scenarios and the setting with minimal production related costs, such as material cost, labor cost and so on was selected. They claimed that if more than one platform was desired, only adjacent products would share the platforms. After studying different demand and platform-to-variant assignment combination, the authors found that a single platform was

satisfactory in cases of uniform demand distribution, but more platforms were advantageous when significant demand gaps created a distance between low and high capacity product variants. They concluded that the number of platforms was related to the product performance gap, which means the desirability of multiple platforms increases when the performance gaps are great.

Following their previous work, a successive research that considers evolving family of products was conducted by Seepersad et al. (2002), using the aggregated utility function deviation instead of component performance deviation from their targets. Seepersad et al. (2000) didn't take into account the effects of customer valuation, while Seepersad et al. (2002) used utility functions to reflect the customer preferences to some extent. Seepersad et al. (2000) studied a future market in the context of uncertain product requirements. They reached a similar result with Seepersad et al. (2000): in a widespread market, multiple platforms would bring more benefit.

Both Seepersad et al. (2000) and Seepersad et al. (2000) have tried to optimize both platform and individual products simultaneously, and determine the optimal number of platforms from technical and economic views by comparing the results after optimization. However, the optimal number of platform and variant assignment are not determined in the optimization procedure but predefined for the interest of simplicity.

Weck (2005) formulated the multi-platform problem as a weighted least square optimization problem. Their formulation studied all the possible platform scenarios between two extremes: non-platforming, meaning each product can serve as a platform, and single-platforming, in which all the product variants share a same set of design variables. The problems were first solved as a single platform, and subsequently the

number of platform are added by one till the non-platforming scenario. Heuristic algorithms, such as Simulated Annealing (SA), were used to solve this multi-platform problem. At the same time, the variant assignment problems are solved. In addition, net present value (NPV), was used to evaluate the efficience of each multi-platform scenario and determine the optimal architecture of the platform. After fathoming all the platform scenarios, the scenario with the lowest NPV value was selected as the optimal solution.

Weck et al. (2003) put forward a two-stage method to determine the optimum number of product platforms to maximize overall product family profit. Like Weck (2005), Weck et al. (2003) iteratively determined the optimal number of platforms N from 1 to M (the number of market segment, different from the number of products by Weck (2005)), and the optimum platform architecture was same with the platform architecture for market leader in the market segment (which were assumed known as a priori). At the first stage, the distance of platform design variables from market leader was treated as an objective function with the number of platform N as a constraint (N starts from 1 to M). At this stage, the optimal number of platform was determined and all platforms were assigned to appropriate market segment. Then the product variants were optimized.

Comparing to multiple platform formulation by Seepersad et al (2000&2002), Weck et al (2003) and Weck (2005) made a great progress in solving the platform positioning problem when optimizing the design for each platforming scenario. However, their methods are time-consuming and cumbersome because they solve the problem M times (M is the number of platform scenarios).

A most recent research about multiple platform design is conducted by Dai and Scott (2007). In their research, the optimal values for design variables are reached by

maximizing the aggregated performance objective functions. They used clustering analysis to decide the number of platforms, and the platform structure (which design variables should be platform variables) and the distribution of each product variant are determined by grouping the "close" enough design variables which have been optimized for individual product variants. Then the platform strategy is optimized and platform and differential variables are got by maximizing the performance objective. If the performance loss for individual product due to commonality is not allowable or tolerable, the platform architecture should be redesigned. In their method, the number of platforms is heuristically determined and the designer can adjust the clustering strategy according to the commonality level and requirements from market. Though the method used in their method looks promising, they only considered the performance requirements and the lack of the analysis the cycle-time analysis about the platform design.

Reviewing all the above literature about multiple platform optimization problems, the researchers did not get the optimal number of platforms during the optimization procedure. They got the optimal number of platforms either by enumerating all the possible numbers with predefined platform-to-variant assignment (Seepersad et al. (2000), Seepersad et al.(2002) and Weck 2005) or by heuristically searching the number when the objective target has been met (such as Dai and Scott, 2007, Simpson et al. 2005, Weck et al. 2003). These methods don't ensure the optimality of platform number or need great effort to reach optimality. This limitation is solved by Hernandez et al. (2002) through a hierarchic product platform method: Product Platform Constructal Theory Method (PPCTM). This method combines hierarchical systems theory and constructal theory to realize a multi-stage optimization method. After solving the lower level

problem, the calculated values are used as the given value for upper level till the highest level. The shortcomings of this method are that 1) the number of stages is artificially decided by the designers, which may influence the optimal values and 2) the method is not intuitive enough, though the optimal number of platforms are addressed and reached during the optimization procedure.

In the economic view, since manufacturing costs and market demand greatly influence decisions relating to platform extent, the existing research does not fully consider their effects (such as Dai and Scott, 2007) or partially considers them by simplified assumptions (such as Seepersad et al. 2000 and Seepersad et al.2000). Fortunately, the last limitations have been noticed and studied by researchers such as Williams et al (2004 & 2007) and Kulkarni et al. (2005). Williams et al (2004 & 2007) and Kulkarni et al. (2005) developed their methods based on PPCTM by Hernandez et al. (2002) and included uncertain market demands into the platform extent problem, which filled the gap of Hernandez et al. (2002) where they only considered technical issues and ignore the external factors. However, all of them failed to consider the life-cycle issues in their models which may definitely influence their optimal decisions.

2.5.3 Summary and Discussions

Multi-platform strategy has been studied and reviewed from both management-centric and engineering-centric views. According to the three research questions in Section 1.2 that will be addressed in this research, various multi-platform models are summarized in Table 2.2.

Simpson et al. (2001) mentioned two product family realization approaches: Top-down and Bottom up. Top-down approach means the company strategically manages and develops a family of products based on a product platform and its derivatives. In this method, the product variants do not necessarily exist. The second method, bottom-up approach, means the company redesigns or consolidates a group of distinct products to standardize components to take advantage of platform strategy. The top-down approach is more suitable for the evolving family of products, which faces the future market, while bottom-up approach is used to implement platform strategy for existing products. Table 2.2 indicates which research questions are addressed and whether a top-down or bottom-up approach is used in the various multi-platform models described in the literature.

In summary, many researchers have claimed that multiple platforms will bring benefit for companies, but they don't clearly specify environments in which the multiple platforms are necessary and will bring economic benefit for industries, even though Seepersad et al (2000), Seepersad et al (2002) and Dai and Scott (2006) and Sahin (2007) mentioned that the multiple platforms are favorable when the gaps between product variants are great. In addition, in some papers, such as Seepersad et al. (2000, 2002), the number of platforms have been determined before the optimization of the each platform scenario, and in other articles, such as Dai and Scott (2007) and Weck (2005), the heuristic method is used and makes the optimization methods for multiple platforms quite complex and time-consuming. Thirdly, the ultimate goal of platforming is long-term profit maximization for companies; unfortunately, very little work focuses on addressing how much profit the multiple-platform strategy can bring.

Table 2.2: Multiple-platform strategy literature classification

		Q1	Q2	Q 3	Top-down	Bottom-up
Management centric	Martin and Ishii (2000)	+			+	
	Martin and Ishii (2002)	+			+	
	Kumar and Allada (2005)	+	+	+	+	
	Sahin (2007)	+	+	+		+
	Weck et al. (2003)		+	+	+	
Engineering centric	Seepersad et al. (2000)	+	+	+		+
	Seepersad et al. (2002)	+	+	+		+
	Dai and Scott (2007)	+		+		+
	Weck (2005)	+	+	+		+
	Hernandez et al. (2002)	+	+	+		+
	Williams et al (2004 & 2007)	+	+	+		+
	Kulkarni et al. (2005)	+	+	+		+

+ indicates the literature addresses related questions and/or methods it uses

2.6   Cost Estimation Techniques

Product cost estimation approaches have been broadly explored by researchers. The existing cost estimation approaches can roughly be classified into two categories: qualitative (such as case-based methodology, such as Rehman and Guenov (1998) and Ficko et al. (2005)) and regression analysis models (such as Hundal (1993), Poli et al. (1988), Lewis (2000) and Pahl and Beitz (1996)) and quantitative estimation (such as parametric cost estimation by Cavalieri et al (2004) and Hajaro (1998)). The qualitative approaches are mainly based on analyzing a large amount of historical data to find out the empirical or statistical relationships between costs and design variables in production processes, while the quantitative approaches can build upon analytical relationships

between cost and design variables (i.e. materials, weights, etc). More comprehensive review can be found by Niazi et al. (2006).

Park and Simpson (2005) divided existing cost functions into (1) the empirical relationships between design variables and production processes based on historical and operating data and past experience (2) engineering relationships between design and process variables including economic factors and (3) statistical relationships between design variables and process characteristics. These cost functions are, however, developed by assuming that the costs are related to one (or more) of the design variables even if some of the production costs are not related to the design variables. These methods try to build up the relationship between design specification and cost information and work well for the direct cost estimation.

In addition, the methods mentioned above are more suitable to estimate the cost in individual products rather than multiple products. The shared overhead costs are still kept intractable for individual products. To estimate the cost in product family development, where there are some product parts or production processes sharing, the methods mentioned are not applicable if there are no adjustments.

Activity-based Costing (ABC), which is good at indirect cost estimation by tracing the resource consumption in the production process, has been studied to provide a inspiring way to estimate the indirect cost in the product family design. ABC is implemented by hierarchically decomposing indirect costs into engineering-related activities and recording the resource consumption consumed by individual products. In general, production costs are generated by production activities from material procurement to product distribution, and these activities will consume labor and other resources. In ABC,

activity cost drivers are used to describe the way in which resources are consumed and the quantitative relationship between cost and activities are linear. The general procedure to implement ABC method is (Hilton et al. 2000): Activity identification, cost driver identification and rate calculation, cost driver consumption for new product, and total cost calculation.

Activity-Based Cost (ABC) (Cooper and Kaplan 1991, Hundal 1997, Ben-Arieh and Qian 2003) estimation techniques have been used to efficiently estimate cost incurred in product family development. Traditional accounting systems focus more on direct cost such as material costs and labor costs, which are traceable to products, and focus less on indirect costs such as overhead costs, which are shared by more than one product. Siddique and Repphun (2001) and Park and Simpson (2005) have applied ABC techniques to investigate the cost saving caused by platform-based product development strategy. Their models could identify indirect costs consumed by individual products by hierarchically decomposing indirect costs into engineering-related activities. It has been noted that ABC can more accurately identify the indirect cost due to components or subsystem sharing and address non-value-added activities to reduce or eliminate them (Park and Simpson, 2005). However, the existing ABC does not show the learning effects caused by the mass production of platform components.

2.7   Summary of Chapter 2

In this chapter, integral and modular product architecture are discussed with regard to platform planning and managing. Information management tools that are widely used in product design, especially in product family design, are briefly introduced to illustrate

how information transmission and conversion between customers and engineers are made and the metrics of assessing product platform are presented. The decision management tools in product design, such as QFD and DSM, and their extensions in product family development are provided. The models that address the customer requirements are discussed in the platform planning and product variety development. The optimization models (including multi-objective optimization models) related to determining the values of the common and unique design parameters are categorized and compared. Lastly, ABC engineering cost estimation approach is discussed.

Based on the previous work by others, in Chapter 3, the new methodology that investigates the multi-platform strategy is put forward and the problems presented in Section 1.2 are addressed.

Chapter 3 : Approach and Method

In this Chapter, a new method is put forward to identify a multi-platform strategy, including determination of product platform planning, assignments of platforms and products, specifications of the values of design variables, and the determination of an optimal platform strategy. It is assumed that there is a set of differentiated products with specific design features that the company wants to combine into a platform-based product development strategy in order to gain the advantages that result from a multi-platform strategy. In the sections that follow, an overview of the proposed method is first described. Details of the method are then provided for each step along with applied tools and techniques.

3.1   Methodology Overview

The method in this research takes a bottom-up approach. This coincides with the approach that companies take to redesign or consolidate a group of distinct products to standardize components in order to take advantage of platform strategy.

The framework of the multi-platform strategy is shown in Figure 3.1. As shown, the research method consists of 3 phases, including multi-platform structure identification phase, product family optimization phase and platform alternative evaluation phase. The primary focus is to support a product design and development team as they go through analyzing the market segments and identifying the common customer requirements to determine the most promising platform strategy. For each phase, there are several steps to achieve/accomplish the phase tasks. On the left of the figure, the major inputs are displayed for the research method to generate expected outputs: potential platform

setting, design specification, and the most promising platforming strategy. The right of

the figure presents several tools and methods that are applied to support decision making

and mathematical programming.

Figure 3.1: Methodology procedure and tools
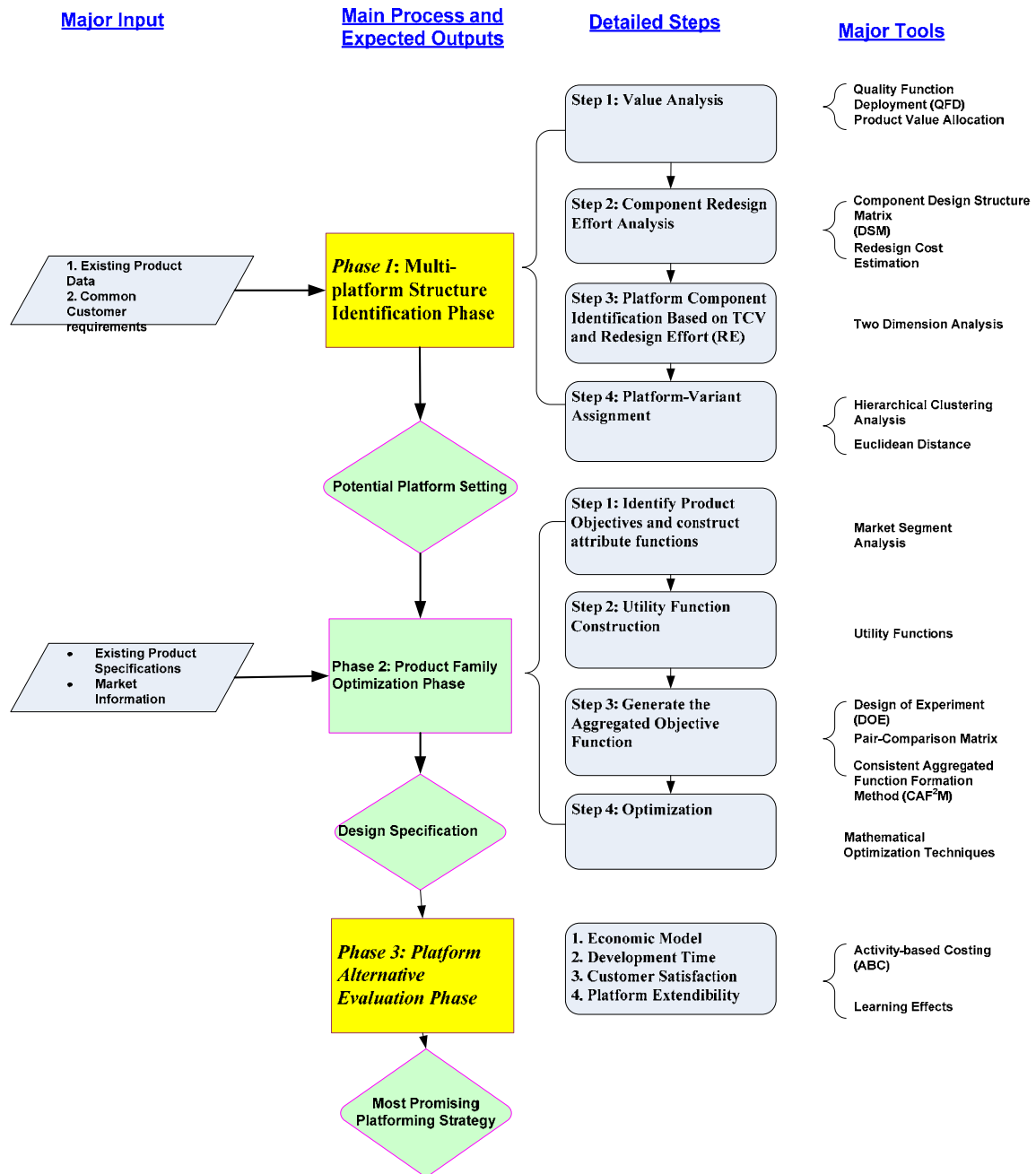
Phase 1 starts with the common customer requirements, maps the product values to product components through QFD and identifies the least value-added components. Meanwhile, the redesign effort analysis is conducted to assess the potential redesign efforts to determine the components with less redesign efforts. Then the platform components are determined with regard to component values and redesign efforts. After that, the clustering analysis is applied in the platform components to flexibly and intuitively determine the multi-platform strategy alternatives based on the product similarity/commonality. The clustering analysis also specifies how to derive product variants based on the appropriate platforms (product-platform assignment).

Phase 2 specifies the product details through a mathematical optimization model. To satisfy multiple goals of a family of products, multiple engineering attributes are treated as objectives and utility functions are adopted to reflect the product positioning preference and address the interior competition in the family. To reduce the problem complexity and take advantage of existing research on single-objective optimization techniques, the multiple objective optimization problem is converted into a single-objective problem by conducting the Consistent Aggregate Function Formation method (CAF^{2}M), which is derived from DOE (to identify the significant factors through statistical analysis) and AHP (to specify the relative importance of experimental scenarios and check decision making consistencies). Design specifications are finalized by optimization techniques, depending on the property of the mathematical model (i.e., linear or non-linear, convex or non-convex).

Phase 3 is the platform strategy evaluation phase, and various metrics, such as engineering related cost, development time and so on, can be considered as decision

attributes. Among these metrics, economic benefit is a big concern of the manufacturing company. The essence of product family design and development is to gain the benefits from economy of scale as well as to offer sufficient product variety. An adjusted Activity-Based Costing (ABC) model is implemented to trace the economic contribution due to centralized product management and mass production. In addition, the indirect costs are allocated to individual products by tracing production-related activities. Learning effects, which are difficult to address in traditional accounting systems, are also considered. The adjusted ABC model offers a means to access the individual product cost in an environment where production resource sharing occurs among multiple products. The three phases and associated sub-steps are demonstrated in the following sections.

3.2   Phase 1: Multi-platform Structure Identification Phase

The proposed method for identifying product platforms is described in this section. The multiple-platform structure is shown in Figure 3.2. For each product in Figure 3.2, there are four slots to fill, including ComSlot 1, ComSlot 2, ComSlot 3 and ComSlot 4. Products in the family are composed of four product elements. These elements can be product subsystems, components, parts, or modules. For simplicity, it is assumed that products are constituted from product components. For each slot, there are several instances with different performance levels. $Ins(i,j)$ designates j^{th} instances for component slot i. Product variants are realized by adding component instances into the corresponding slots. In each slot, only one instance can be filled at a time to realize a product variant.

Figure 3.2: A multi-platform structure
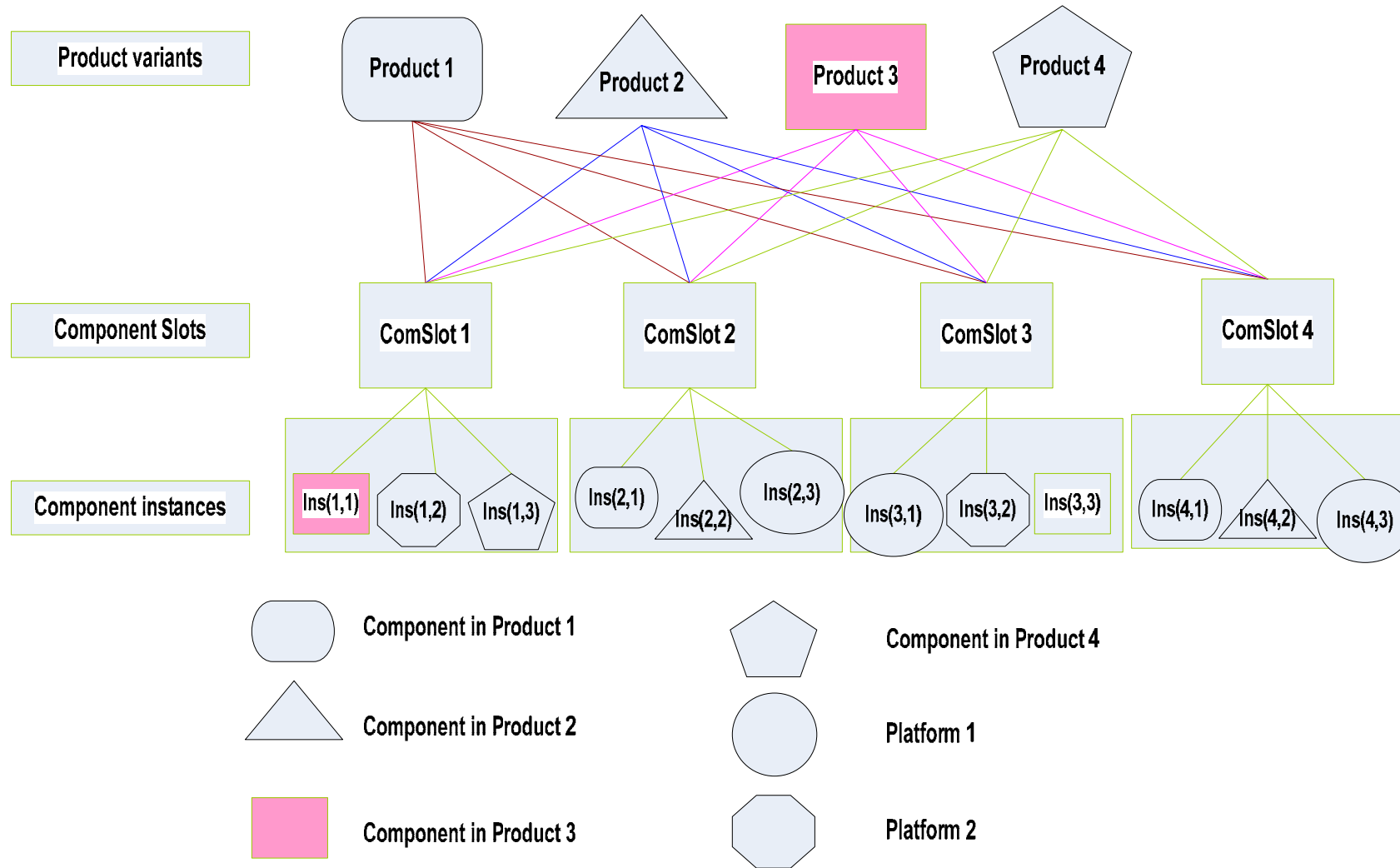
For example, Product 1 is derived from component instance (1,2), (2,1), (3,2) and (4,1) and Product 2 includes component instance (1,2), (2,2), (3,2) and (4,2). From the definition of a platform, if two products have the same component instance for the same component slot, the instance is a platform instance, and the component is a platform component. For example, Component instances (1,2) and (3,2) are shared by Products 1 and 2. Therefore, they form one platform for Product 1 and 2. In Figure 3.2, there are two platforms: Platform 1, shared by Product 3 and 4, and Platform 2, shared by Product 1 and 2. There are multiple platforms serving products in one family, and this is referred to a multi-platform strategy.

In this phase, the platform elements and unique elements need to be determined along with the instantiation of product variants with platform element combinations. There are four steps to complete this phase as shown in Figure 3.3. These steps include:

Step 1: Allocate product values into product physical elements using the Quality Function Deployment (QFD) method, starting from the Common Customer Requirements (CCRs);

Step 2: Identify the redesign efforts for the elements in the product variants;

Step 3: Identify platform elements with low value-added and redesign efforts; and

Step 4: Platform-Variant assignment.

The inputs of this phase are common customer requirements (CCRs). The CCRs are the requirements that are expected by customers in more than one product. For products in a family, individual products have some common elements and properties, and customers have similar expectations about product functionalities. These commonalities/similarities

ensure that platform-based product development is a good strategy for companies to gain the economy of scale and scope, cost reduction and short development time.

Figure 3.3: Flowchart of platform identification method
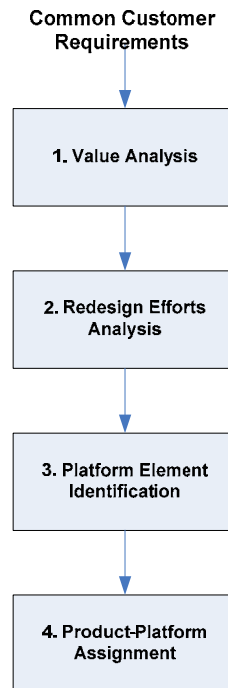
The platform setting is identified based on these product similarities. Platform-based products do not necessarily have the same values for design variables, but are required to achieve similar functions, activities or service. Since a platform is not only serving the current market, but several product generations, the common customer requirements are not only from current customers and market, but also anticipated for the future. How to collect CCRs is not the focus of this research, but a method put forward by Kurtadikar et al. (2004) can be applied, which includes: evaluating individual products for their requirements and record, listing all the requirements and asking for re-evaluation, and calculating the frequency of each requirement. The CCRs are mapped into Engineering Attributes (EAs) by QFD, which can describe the products engineering characteristics.

EAs can be functional descriptions or technical specifications. Greater detail for each of the four steps is provided in the following sections.

3.2.1 Step 1: Value Analysis

The goal of this step is to allocate the subjective product values of individual products to engineering attributes (EAs) and then to the physical elements. The products in the family focus on different market segments and each market segment has its own key EAs, which characterize their uniqueness and attract customers' attention. Within a product family, these key EAs have different feature levels in the market, and also differentiate the products from each other. Some EAs are thought to be more critical in a product while less important in another product. For example, for a water cooler family, customers think EA "cool down time" is more important than "energy saving" for office usage, while energy saving would be a critical factor for home-usage water coolers. These importance differences indicate different product values for EAs in products.

 The EAs are realized by the product physical elements. As such, the instances that fill in a slot (refer again to Figure 3.2) play different roles in different products. Therefore, on the physical level, the physical elements have different allocated product values. Fujita et al. (2000) applied a QFD-based approach to determine cost-worth analysis for a product family. In this research, a two-phased QFD approach is used to hierarchically allocate product values to engineering attributes (EAs) and then to physical elements.

Phase I-QFD is to identify the importance of EAs in product variants and these importance values help to differentiate products in a family. An Attribute Significance Matrix (ASM) is accomplished by generating a weighting factor with a 1-9 rating scale:

1-very weak, 3-weak, 5-medium, 7-strong, 9-very strong, and scores of 2-4-6-8 are intermediate values. A Relative Attribute Significance Matrix (RASM) is derived by normalizing values in ASM. Phase II-QFD is applied to map each EA to a set of product elements. The correlation matrix between EAs and product elements is built up and the values in the matrix indicate how much (e.g. strongly/weakly) the product elements affect the realization of EAs. After normalization, a Relative Correlation Matrix (RCM) is obtained. Phase I and Phase II QFDs are shown in Figure 3.4.

The allocated product values in product elements for each product are calculated as follows:

$$V= RASV^T*RCM \tag{1}$$

Total Value (TV) of a product element is the sum of allocated product values across products in the family. Based on the study by Thevenot and Simposon (2004) and Park and Simpson (2006), the platform is composed of components that give low values to customers. Therefore, elements with less TVs are more suitable to be platform elements. Consequently, the loss of the product values will be relatively lower while sharing the platform components.

Figure 3.4: Element values and redesign efforts identification procedure
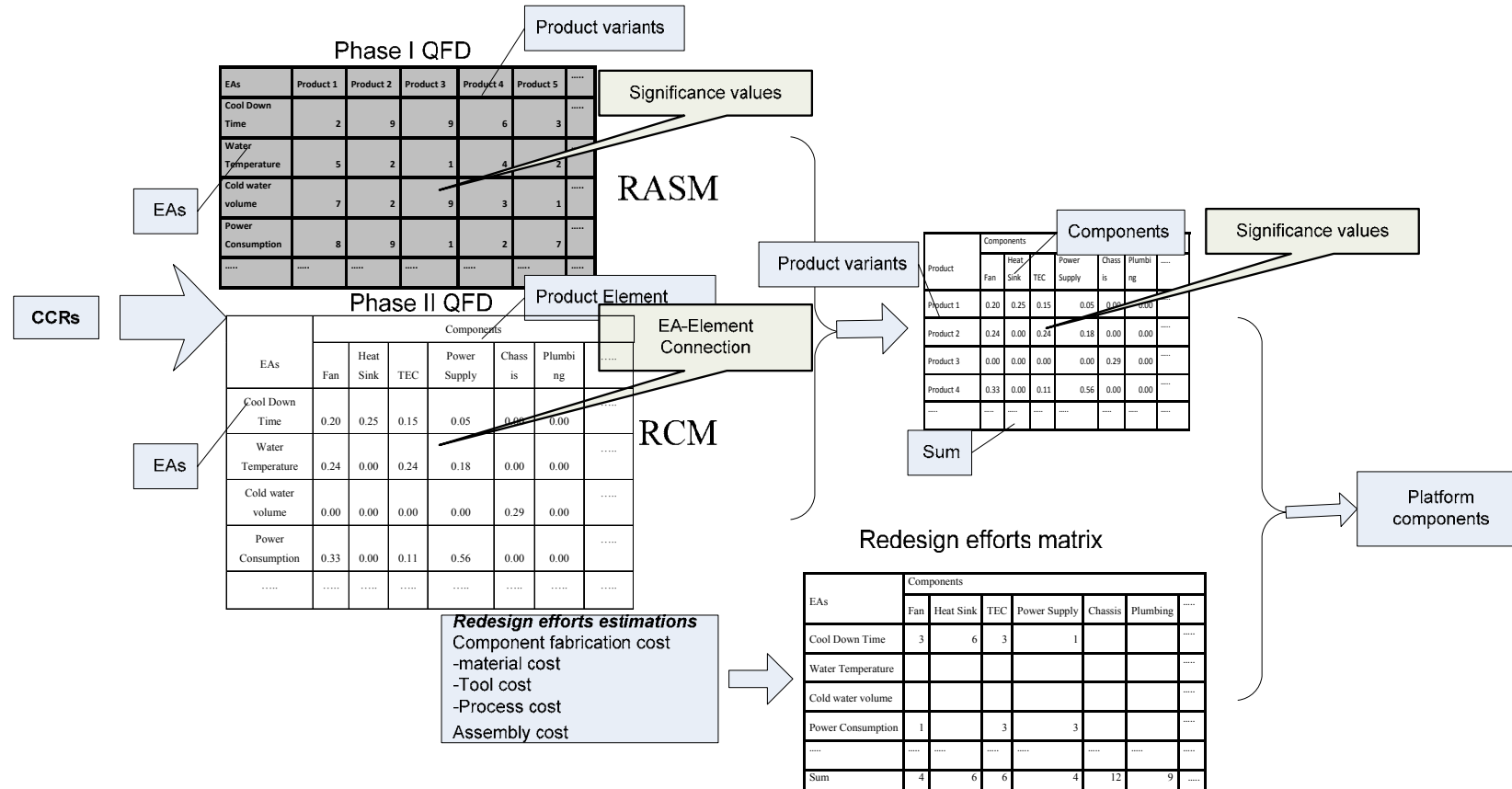
3.2.2 Step 2: Redesign Effort Analysis

Since this research method is intended to implement a platform-based strategy on existing products, the execution of a platform strategy will definitely cause product redesign due to the compromise between products. The redesign efforts are required to realize target EAs as well as product element sharing and reuse. Furthermore, a platform strategy is also concerned with product generational service; hence, the design team must determine how long the platform will last. Therefore, the possible future adjustment should be taken into account along with the consideration of redesign efforts.

The target EA values could be based on information from conjoint analysis, trend analysis, expected new markets, or expected competitor introduction of products. They are obtained on the analysis of previous trends and marketing data. In addition, since platforms are aiming to support not only one generation of products, future factors, such as future customers preferences, potential competitors and new regulations, are involved to determine the targeted EA values. Formal methods, such as Yu et al. (1998), give a more detailed approach to estimating future target values.

In addition, product elements may be dependent on each other, thus, the redesign of one part/component may cause corresponding changes of other related parts/components. The dependency between product elements can be identified by the design structure matrix (DSM). DSM is a matrix-based representation of a system or a project. DSM contains the constituent subsystems or activities and corresponding information exchange and dependency (Alizon et al., 2006).The redesign efforts of one component caused by another component are counted in the redesign efforts of the originating one. The components redesign efforts are determined by estimating their redesign costs, which are

consumed to realize targeted engineering attributes. These costs include fabrication cost (material cost, tool cost, and process), assembly cost, etc.

Since the accurate cost information is not available until the products are produced, the redesign efforts of product parts/components are estimated by the design team through their knowledge and experience. These costs are roughly expressed as a percentage of the original cost to design and are expressed by qualitative values (Martin and Ishii, 2002). The detailed description of redesign rating is shown in Table 3.1.

Table 3.1: Redesign rating description (Martin and Ishii, 2002)

Redesign rating	Description
9	Requires major redesign of the component (>50% of initial design costs)
6	Requires partial redesign of component (<50%)
3	Requires numerous simple changes (<30%)
1	Requires few minor changes (<15%)

According to Martin and Ishii (2002), product elements with fewer redesign efforts would be more likely to be potential platform elements. Intuitively, if the redesign efforts are too high or beyond the redesign budget, then the redesign activities will not be conducted. Definitely, the companies would prefer to use less redesign efforts to launch the product platform and a group of individual products, and further obtain the economic benefits due to mass production. The redesign effort level of each component is determined by the engineering team's expertise and judgment. The engineering team knows the relationship between EAs and product parts/components and then estimates the level of redesign effort if the elements are redesigned.

Product elements are evaluated by assigning a redesign rating through the cost analysis for achieving corresponding targeted EA values. In Table 3.2, "Chassis" in a water cooler requires the greatest redesign effort, and "Fan" and "Power supply" need the least redesign efforts in a water cooler. Components in different products need different levels of redesign efforts to reach their specific targets. The platform component identification is based on group analysis of the family, not just individual products. Total component redesign efforts are calculated by linearly aggregating component redesign rating across the family.

Table 3.2: Redesign Effort Matrix for a Water Cooler

	Product Elements						
	Fan	Heat Sink	TEC	Power Supply	Chassis	Plumbing	…..
Cool Down Time	3	6	3	1			…..
Water Temperature							…..
Cold water volume							…..
Power Consumption	1		3	3			…..
…..	…..	…..	…..	…..	…..	…..	…..
Sum	4	6	6	4	12	9	…..

3.2.3 Step 3: Platform Identification Based on TV and Redesign Effort

The sharing elements for the whole product family are determined based on the redesign effort analysis and element value analysis. Ideally, the parts/components with the least redesign efforts as well as the least allocated values are the best platform options (Utopia solution). However, the ideal case does not happen all the time. The platform elements need to be identified by balancing the redesign effort (cost) and allocated values (worth).

 For instance, the redesign efforts and total product values are plotted in Figure 3.5 for the water cooler family. Based on the platform selection threshold specified by the design

team, the points that qualify the threshold values are selected to be the platform components. As shown, the components at the lower left corner are more possible to be the sharing components because the components in that area need less redesign effort and will lose relatively lower differentiating values.

Figure 3.5: Redesign effort-Total product values
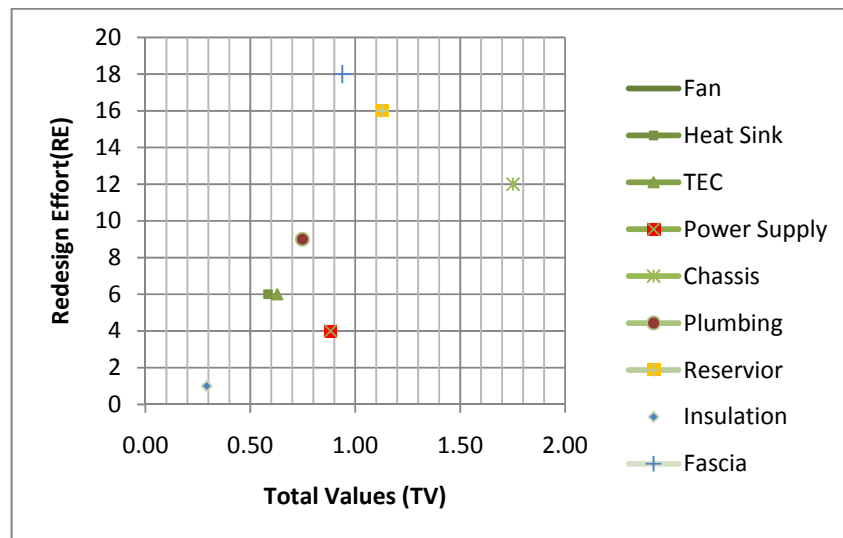
3.2.4 Step 4: Platform-Variant Assignment

In Figure 3.2, each platform component has several instances, which are corresponding to the product variants in the family. The existing products in the family are realized by adding instances into corresponding slots. Product elements are described by a set of design variables. A design variable is any quantity or choice directly under the control of the designers and can characterize the elements. For example, design variables can be geometry parameters (height, width, and length), material (steel, plastics), or mass. To avoid too much compromise resulting from the platform strategy, the instances for platform slots in existing products are separated into several groups, and the products falling in a group will have the same instance for the slot. This instance is named a

platform instance with regard to the platform element. This means for each platform element, there will be multiple instances to fill in the corresponding slots and the sharing only happens among the products that are similar enough. As compared to the single-platform strategy, the compromise in the multi-platform strategy is reduced. In this way, the platform parts/components are less stretched than those in a single-platform strategy, avoiding platform overdesigned or over-stretched to some extent.

The products with the same platform instance are grouped into a cluster with respect to a platform element. Clustering approaches, which are based on similarity or commonality of products attributes, have been investigated in product family design for various purposes. Stadzisz and Henrioud (1995) cluster products based on geometric similarities to decrease product variability as well as reduce assembly complexity and Shirley (1990) describes a clustering process to form single-platform and determine the design specifications across the family. In this research, the clustering approach is applied to each platform element to generate more than one component instance for the corresponding platform element, based on the information from existing products in the family.

Similarity (dissimilarity) of components (which are described by a set of component design variables) can be measured by the metric distance, such as Euclidean Distance or Angular separation (Romesburg, 1990). With the measured similarity (or dissimilarity), clustering analysis method is used to help to classify the products into k subsets, according to their platform component similarity and design team's judgment. The values of k can be flexibly determined according to the product similarity. Hierarchical clustering analysis or K-means clustering analysis approach is applied to divide the

existing component instances into k subgroups and the instances in a same group will be standardized. In this way, the component variability as well as the total variations across the product family is reduced. Meanwhile, since the component instances are corresponding to specific products, the product-platform assignment problem, which specifies which product using which platform, is determined in the clustering procedure.

3.3   Phase 2: Product Family Optimization Phase

In the multi-platform structure identification phase, the platform setting alternatives have been determined according to redesign efforts and element values. The design team can flexibly determine the number of platforms by setting the number of clusters in the clustering analysis. The customer requirements have been defined and mapped to hardware or software components in the first phase, and the design vectors also have been defined, but their specific values for design variables x_i's are still unknown. The optimization phase is to specify the values of the design variables to realize the whole product family with specified technical objectives and performance constraints. Multi-objective optimization models are formulated to parameterize the values of platform and non-platform design variables. Four steps are included in this phase:

Step 1: Identify primary attributes;

Step 2: Utility function construction;

Step 3: Generate the aggregated objective function; and

Step 4: Optimization.

3.3.1 Step 1: Identify Primary Attribute

Products in a family target diverse market segments and fulfill various customers' requirements for particular markets. Different markets have different target performance levels for the corresponding products. The products in the family are positioned in different market segments to address various customer needs, thus the market segment analysis is required to support product positioning and avoid internal and external competition. The internal competition means the products in a family will compete with each other since they own some commonalities among them and the external competition indicates how a company competes with its competitors.

Since multiple products are planned simultaneously, it is natural to consider several goals to achieve. Therefore, in the optimization model, multiple objectives will be identified for the whole family according to the distinguishing characteristics of the market segments and the corresponding product primary attributes. Primary attributes in this research are the performance attributes that can be used to generate objective functions in the optimization models. They are defined as a set of functional attributes that can quantify the product's main value-delivering functions and differentiate market niches. For example, for automobiles, its primary value-delivering function is transporting passengers and cargo comfortably, quickly and economically from origin to destination. The functional attributes corresponding to the main function are passenger volume (PV), cargo volume (CV), towing capacity (TC), fuel economy (FE) and acceleration (AC) (Weck, 2006). The functional attributes that are chosen as objective functions are referred to as objective attributes and denoted as f.

In addition, the same attribute in different product variants in a family may target different performance values. For example, the target value of PV for a mid-size sedan is 101.7 cft while for a large-size sedan is 105.6 cft. In addition, simple minimal or maximal values are not enough to represent the specific attribute requirements for product variants. How to display various requirements of functional attributes is discussed in the following sections.

3.3.2 Step 2: Utility Function Construction

As mentioned above, instead of obtaining extreme values for all attributes, in some cases, an intermediate value or interval may be more preferred in the context of product family optimization. To represent the unique properties in product variety optimization, utility functions are utilized. Based on previous work by Messac et al. (2002), the primary attributes could be classified into two groups: hard and soft. Hard attributes indicate that the requirements must be satisfied without violation, and can be included in the constraint set. Soft attributes indicate that the performance attributes are acceptable with flexibility. The acceptance levels are expressed by utility values, which lie in the range of 0-1.

The objective functions in the multi-objective optimization model are established upon soft attributes. Four types of utility functions are established to represent four kinds of attribute preferences respectively, shown in Table 3.3. As known, the larger utility value, the closer to its targeted value. For example, for the attributes in soft group 1: "Smallest-is-Best", the best value is the minimal value. Thus the utility function for attributes in soft group 1 is a decreasing function and utility value equal to 1 when the attribute value reaches its minimum, subject to constraints.

Table 3.3: Attribute classes

	Soft	Hard
1	Smallest is best (i.e., Minimization)	Must be smaller (i.e., $f \leq f_{max}$)
2	Largest is best (i.e., Maximization)	Must be larger (i.e., $f \geq f_{min}$)
3	Value is best	Must be equal (i.e., $f = f^*$)
4	Range is best	Must be in range (i.e., $f_{min} \leq f \leq f_{max}$)

Another reason for the application of utility functions is to make the attribute within the commensurate scale (0-1 interval), then a unitless aggregated objective function (AOF) can be formed. Table 3.4 shows the soft objective attributes and the corresponding utility function description.

Table 3.4: Soft primary attributes and utility functions

	Soft	Utility function description
1	Smallest-is-Best (i.e., Minimum)	A decreasing function
2	Largest-is-Best (i.e., Maximum)	An increasing function
3	Value-is-Best	A triangular function with utility value 1 at peak
4	Range-is-Best	A trapezoidal function with utility value 1 in the range interval

Four types of utility functions are illustrated in Figures 6-9. Other functions, such as exponential and quadratic functions, may be more suitable to express the decision makers' opinion, but in this section, only linear functions are used to demonstrate these four types of utility functions.

The Largest-is-Best type utility function is shown in Equation (2), and plotted in Figure 3.6. It is used when the goal value of the attribute is set at the maximal values (f_{max}). The closer to the maximal values, the higher the utility value.

$$u(f) = \begin{cases} 0, & f \le f_{min} \\ \frac{f - f_{min}}{f_{max} - f_{min}}, & f_{min} \le f \le f_{max} \\ 0, & f > f_{max} \end{cases} \qquad (2)$$

Figure 3.6: Largest-is-best Type Utility Function
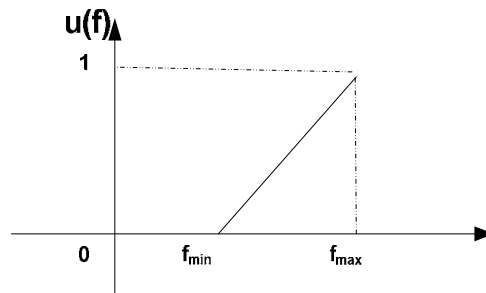
The Smallest-is-Best type function is shown in Equation (3) and plotted in Figure 3.7. It is used when the goal value of the attribute is set at the minimal values (f_{min}). The closer to the minimal values, the higher the utility value is.

$$u(f) = \begin{cases} 0, & f \le f_{min} \\ \frac{f_{min} - f}{f_{max} - f_{min}}, & f_{min} \le f \le f_{max} \\ 0, & f > f_{max} \end{cases} \qquad (3)$$

Figure 3.7: Smallest-is-better Type Utility Function
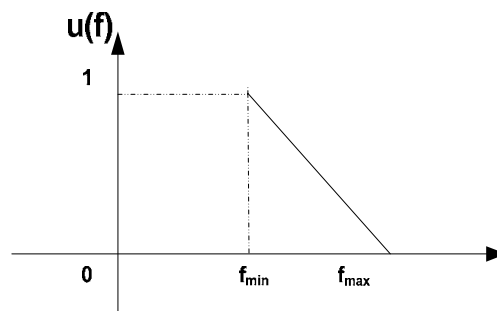
The Value-is-Best type function is shown in Equation (4) and plotted in Figure 3.8. It is used when the goal value of the attribute is set at the intermediate values. In Figure 3.8, the utility value 1 happens at $f_{D\text{-val}}$ point.

70

$$u(f) = \begin{cases} 0, & f \leq f_{min} \\ \dfrac{f - f_{min}}{f_{D-val} - f_{min}}, & f_{min} \leq f \leq f_{D-val} \\ \dfrac{f_{max} - f}{f_{max} - f_{D-val}}, & f_{D-val} \leq f \leq f_{max} \\ 0, & f > f_{max} \end{cases} \qquad (4)$$

Figure 3.8: Value-is-Best Type Utility Function
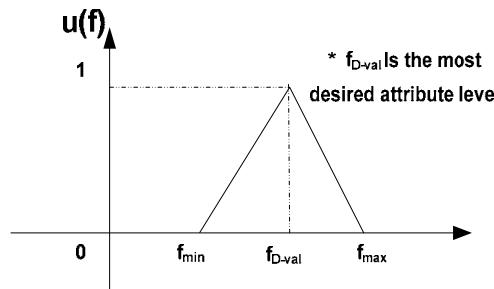
The Range-is-Best type function is shown in Equation (5) and plotted in Figure 3.9. It is used when the goal valuof the attribute is set at the intermediate values.

$$u(f) = \begin{cases} 0, & f \leq f_{min} \\ \dfrac{f - f_{min}}{f_L - f_{min}}, & f_{min} \leq f \leq f_L \\ 1, & f_L \leq f \leq f_U \\ \dfrac{f_U - f}{f_U - f_{max}}, & f_U \leq f \leq f_{max} \\ 0, & f > f_{max} \end{cases} \qquad (5)$$

Figure 3.9: Range-is-Best Type Utility Function
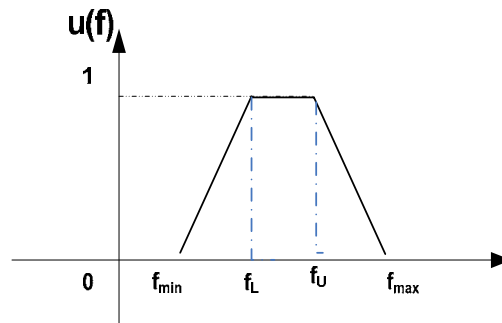
As known, product attributes are subject to a set of physical constraints. Similarly, the constraints are classified into hard and soft constraints. The constraints include the technical constraints and the upper and lower bound of the design variables, as well as commonality constraints. They can be equality and inequality constraints. The products must satisfy the hard constraints, while the soft constraints can be satisfied with some flexibility. For soft constraints, utility functions are built according to the characteristics of the constraints, just as the utility functions for the soft attributes and threshold utility values are set to represent the acceptance level of constraint violation. The decision makers specify the acceptable interval or values for the soft constraints, such as utility values larger than 0.8.

For the soft inequality constraints, the utility functions are one-direction acceptance, such as shown in Figure 3.10, and for the soft equality constraints, the utility functions are two-direction acceptance, such as triangular function shown in Figure 3.11. For both soft inequality and equality constraints, the design team can specify the acceptable lower and upper utility bounds.

Figure 3.10: The Smallest-is-Best utility with acceptable level 0.8
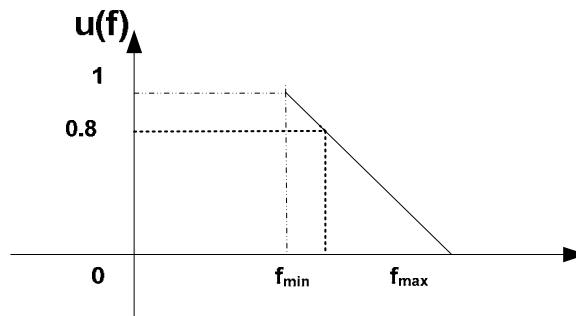
Figure 3.11: The Value-is-Best utility with acceptable level 0.8
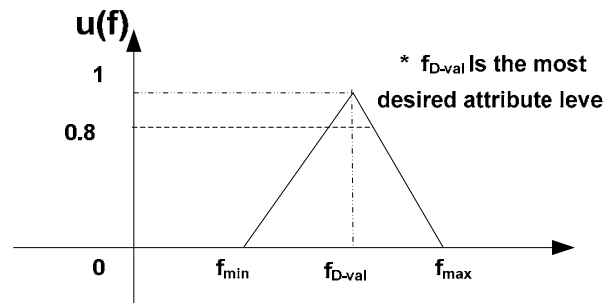
3.3.3 Step 3: Generate the Aggregated Objective Function (AOF)

In Steps 1 and 2, objective functions and constraints are identified, and the parameters to establish the utility functions for objective attributes are unknown. In this step, these unknown parameters will be specified. For an objective attribute, the value interval is confined by the upper and lower values. One approach to obtain the lower and upper bound values for each objective attribute is to minimize and maximize it respectively. The shortcoming of this approach is that the real 'max' or 'min' are not quite possible to reach or even close. Thus, the pseudo-min and pseudo-max are preferred. And they can be obtained by optimizing the objective attributes individually with minimum or maximum, subject to the constraints identified in Step 2. Then the objective matrix will be established, as shown below.

73

$$\text{Objective Matrix} = \begin{Bmatrix} f_1(X_1^*) \, f_2(X_1^*) & \cdots & f_{N-1}(X_1^*) \, f_N(X_1^*) \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ f_1(X_N^*) \, f_2(X_N^*) & \cdots & f_{N-1}(X_N^*) \, f_N(X_N^*) \end{Bmatrix}$$

where, X_j^*: optimal decision variable vector when optimizing the ith objective function; $f(X_j^*)$: the values of ith objective function with X_j^*; N: the number of objective functions.

Identify the upper and lower bounds of each objective attribute from the objective matrix.

$$f_{max}^i = max_{k=1\ldots N}\{f_i(X_k^*)\} \tag{6}$$

$$f_{min}^i = min_{k=1\ldots N}\{f_i(X_k^*)\} \tag{7}$$

To obtain the Pareto optimal in the multi-optimization problem, compromises are inevitable. Minimal and maximal values in Equation (6) and (7) are values that are obtained without consideration of compromises among multiple objective attributes. They can be used to approximate the upper and lower bound space (Guyot, 1996). The Smallest-is-Best and Largest-is-Best utility functions are set with the knowledge of lower and upper bounds.

For Value-is-Best and Range-is-Best types of attributes, the required parameters are specified through various techniques, such as distributing questionnaires, holding interviews, organizing focus group discussion, conducting conjoint and trend analysis. The formal method by Yu et al. (1998) provides a detailed approach to estimating target product attributes with the aid of possibility distribution knowledge and statistical significance analysis.

After the establishment of multiple utility objective functions, one aggregated objective function (AOF) will be formed to convert multi-objective optimization into a single-

objective optimization, which has been widely researched. The AOF function is a function that incorporates multiple objectives into one equation that can characterize the trade-offs among objectives and reflects the preference level of individual objectives. Weighted sum approaches are widely used but have a disadvantage in the fact that compensation is approximated by linear relationship, ignoring the dependency between objectives. The consistent aggregated function formation method (CAF^2M), which is based upon DOE and AHP, is put forward in this research to generate AOF. Another purpose of the application of the utility functions, except expressing design team's preference, is to convert the objective functions into 0-1 range, eliminating the influence of different scales.

The AOF is generated based on group decision making. In addition, the local check makes sure that the individual decision makers make consistent evaluation for the same set of alternatives and the grand consistency check guarantees coherent preferences among the decision makers in a group. The generic procedure is as follows and plotted in Figure 3.12.

1)   Design of Experiment (DOE) phase

    1.1)   Identify criteria that need to be considered in decision making.

The criteria here are corresponding to objectives identified in previous steps.

    1.2)   Arrange level for each factor and design experiments.

For each objective, two values are selected to define two levels. These values are utility values of objectives. The experiment structure is generated through combinations of

objective levels, according to the rules of factorial design (Box, 1978). Each combination of factor levels is called a treatment.

2) AHP phase

2.1) Generate pairwise-comparison matrix for each decision maker (DM).

Ask DM to compare each pair of treatments to obtain their relative preference with scales 1-9. From the Fundamental Scale, 1 expresses that A and B are equally preferred, 3 that the better of the --is moderately preferred to the worse, 5 that the better is strongly preferred, 7 that the better is very strongly preferred, and 9 that the better is extremely preferred to the worse. Intensities 2, 4, 6, and 8 express intermediate values.

2.2) Check local consistency of comparison matrix for each DM.

Local consistency is defined to ensure that each decision maker makes a consistent judgment for the designed experiments. Local consistency checking is conducted by using a Consistency Index (CI), Random Consistency Index (RI) and Consistency Ratio (CR) by Saaty (1980) for each decision maker.

$$CI = \frac{\lambda_{max} - N}{N - 1} \qquad (8)$$

$$CR = \frac{CI}{RI} \qquad (9)$$

where λ_{max} is the largest Eigen value of comparision marix; N is the size of the comparison matrix, i.e., it is the number of objective functions. RI refers to Table 3.5, which is predefined by Saaty (1980).If the Consistency Ratio is less than 10%, it is consistent. Otherwise, the DM needs to adjust the comparison matrix to make it consistent.

Table 3.5: Random Consistency Index (RI)

n	1	2	3	4	5	6	7	8	9	10
RI	0	0	0.58	0.9	1.12	1.24	1.32	1.41	1.45	1.49

2.3) Get priority values, and check the grand consistency.

Grand consistency is defined to show the DMs in the group that they have consistent opinions to the same set of alternatives. The priority values are calculated by normalizing the principal eigen-vector for each pairwise-comparison matrix:

$$A_m P_m = \lambda P_m \tag{10}$$

P_m is the eigen-vector, and λ is eigen-value. The principal eigen-vector is the vector corresponding to the largest eigen-value. Priority value $\overline{P_m}$ is the normalized the principal eigen-vector for the mth DM. Repeat the calculation M (the number of decision makers) times and get the average priority values $\overline{\overline{P}}$ for each alternative ($\overline{\overline{P}}_i$ is the averaged priority values for alternative i).

$$\overline{\overline{p}}_i = \frac{\sum_{j=1}^{4} \overline{P_{i,j}}}{4} \tag{11}$$

Coefficient of variation (CV) for priority values to i[th] alternative are used to show the grand consistency.

$$CV_i = \frac{\sigma_i}{\overline{\overline{p}}_i} \tag{12}$$

$$CV = \frac{\sum_{i=1}^{8} CV_i}{8} \tag{13}$$

σ is the variance of priority values. If CV is too large to accept, this means that DMs in the group do not hold unanimous opinions to the same alternative. Thus the decision making group needs to negotiate to reach a consensus before implementing the next step.

3)   Parameter obtainment phase

3.1)   Perform main effect calculation.

Take $\overline{\overline{P}}$ as the response of DOE, calculate and identify significant effects through DOE procedure. The approach to calculate the main effect is detailed in standard textbooks, Box et al. (1978) for example.

3.2)   Form aggregated functions with respect to factors.

The main effects obtained in step 3.1 are corresponding weights for criteria and their interactions.

Figure 3.12: CAF^{2}M procedure
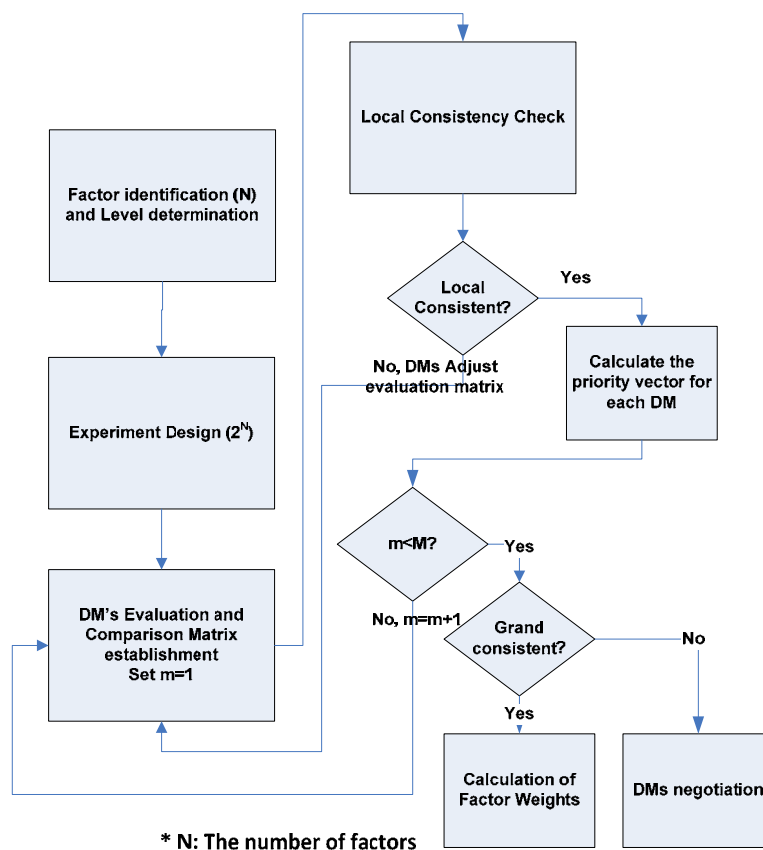
* N: The number of factors

M: The number of Decision Makers (DMs)

Thus the optimization problem is formulated as follows:

$$\text{Maximize} \quad AOF(X) \tag{14}$$

$$\text{Subject to:} \qquad H_i(X) \le 0, \qquad i = 1, \dots n_f \tag{15}$$

$$HG_j(X) \le 0, \quad j = 1, \dots n_h \tag{16}$$

$$SG_j(X) \le 0, \ j = 1, \dots n_s \tag{17}$$

$$x_{a,j} = x_{b,j}, \ a, b \in PI, \ j \in PF \tag{18}$$

$$x_{i,j}^{min} \le x_{i,j} \le x_{i,j}^{max}, \ i = 1, \dots N, j = 1, \dots z \tag{19}$$

Equation (14) is the aggregated utility objective function; $X = [X_1, X_2 \dots X_N] = [(x_{1,1}, x_{1,2}, \dots x_{1,z}), (x_{2,1}, x_{2,2}, \dots x_{2,z}), \dots (x_{N,1}, x_{N,2}, \dots x_{N,z})]$ is the decision variable vector, where X_i is the variable set for product i and N is the number of product variants. Equation (15) is the hard primary attribute requirement that products must satisfy. Equation (16) represents the hard constraints and Equation (17) represents the soft constraints, in which utility functions are utilized. Equation (18) ensures that products that share a platform must have same values of common design variables, where *PI* is the product variant index and *PF* is the platform component design variable set. Equation (19) is the possible range restriction for decision variables.

3.3.4 Step 4: Optimization

In this step, the optimization procedure is performed in order to obtain the optimal values for the design variables. The decision makers (DMs) can then evaluate the optimal results. If the results are not satisfactory, adjustment is inevitable. The adjustment

includes changing the format of utility functions, getting new AOF parameters, relaxing the constraints, etc.

Through Steps 1-4 in Section 3.3.1-3.3.4, one platform setting is obtained. To obtain the Pareto frontier, different AOFs are formed and optimized. The specified new product family is examined as to whether it is satisfied the two criteria in the first phase. If not, the process will be returned to Phase I until the results and design team's desire and estimation match with each other. The optimization steps are repeated for other platform alternatives identified in Phase 1. After specifying the values for all alternatives, the decision making procedure in Phase 3 will be conducted to identify the most promising alternative based on the multiple decision criteria.

3.4   Phase 3: Platform Alternative Evaluation Phase

When the multi-platform structure alternatives are available, engineers should determine which alternative is the most profitable solution for both the company and customers. In this step, the economic factors such as platform-related cost and revenue, and product performance will be taken into account. Also the life-cycle analysis is involved since the platform not only serves for a single generation, but several generations. In this sense, how the platform strategy fits for future market requirements is an important aspect in the solution selection step.

A set of metrics that can facilitate multi-platform evaluation and comparison are needed to support selecting the most promising platform alternative determined in the first two phases.

In this research, a multi-platform assessment framework is proposed based on the existing literature with some appropriate adjustments. The platforms and product family will be evaluated and compared from: cost saving, time reduction, customer attraction (including technical attributes and soft attributes), and platform extendibility. These metrics proposed do not claim comprehensiveness.

3.4.1 Engineering Cost

The manufacturing costs are incurred for tooling and plant equipment, fixed expenditures as well as variable costs such as labor and material. Multiple products are planned and designed simultaneously not only to increase the design complexity, but also to change the cost structure for the individual products in the family. Increasing the number of individual products in the family increases indirect costs relating to inventory, setup, inspection, maintenance, material handling and storage, and rework while the product platform serves as a source of cost savings by reduction of indirect costs as well as direct costs associated with it (Park and Simpson, 2005).

The cost structure for individual products in the platform-based strategy is presented in Figure 3.13. In Figure 3.13, there are two types of product components: platform and non-platform. The platform component is the part shared by several products, while the non-platform component is the part that is designed for a specific product. For the direct cost in both platform and non-platform components, since it is closely related to the engineering specification, it is estimated by costing-by-analogy, which takes the costs of existing products as references and calculates the costs with regard to the references by adjusting for the changes of design variables or other product specifications. The direct cost is assumed to be scaled up or down according to the design variables. The indirect

cost, which is lumped together, will be estimated by implementing an adjusted ABC, which is good at deploying shared costs into individual products by tracing the engineering-related activities. In addition, since the indirect costs are lumped together, the learning effects are included in the estimation of indirect costs.

Figure 3.13: Cost structure
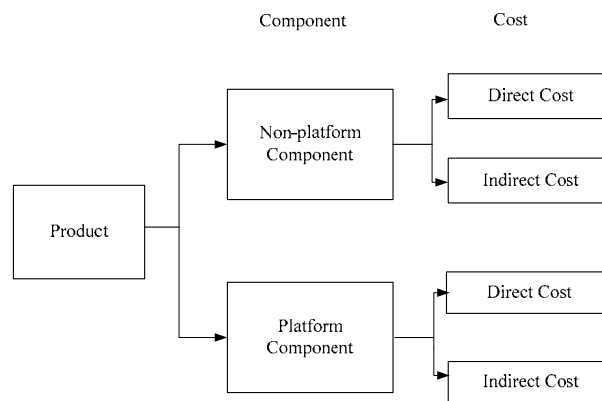
3.4.1.1 Direct cost

Since the direct cost is specifically associated with product design specifications, therefore, costing-by-analogy method would be applicable to estimate the direct cost. Costing-by-analogy takes a known product and its direct cost as a reference and derives the cost of redesign subsystem by proportionally scaling up or down the design variables. The design variables that describe the components are determined by designers based on experiences. The unit direct cost for a component is scaled with respect to a comparative existing product component (a benchmark component):

$$c_D' = \frac{x_1' x_2' \dots x_n'}{x_1 x_2 \dots x_n} c_D \qquad (20)$$

where c_D is the direct cost of the benchmark component; c_D' is the direct cost of the new or redesigned component; x_i is a design variable in benchmark component; x_i' is the design variable in the new or redesigned component.

3.4.1.2 Indirect cost

An important reason for product platforming is the potential cost saving due to the achievement of a significant learning curve. The learning curve manifests itself as progressively decreasing unit resource consumption. An adjusted ABC method is put forward for indirect cost estimation. The adjusted ABC follows the steps in a general ABC approach and also takes the learning effects into account. The procedure to implement the adjusted ABC to estimate the indirect cost for a product family is shown as follows:

1)   Describe the production system based on its activities, and identify the key activities for the indirect costs and the cost drivers for each activity.

Since the platform-based products are still in the design phase, their indirect cost information is based on existing products. Similar products and their cost information are collected. It is assumed that the new products or the redesigned products will have the same production process and resources. An extensive study should be undertaken to identify the production activities. Park and Simpson (2005) used cause-effect diagram to identify possible activity and resource information affected by platform-based product development strategy. These activities are product specific. The identified activities should cover most of the indirect costs. The key activities can be identified by the Pareto principle (20-80 rule: 80% of the effects come from 20% of the causes). For each production activity, its corresponding cost driver is defined and it should be a measurable

unit describing how the corresponding activity consumes the production resources in the production process. For example, setup costs are involved, and the setup time would be an appropriate cost driver for the setup activity.

2) Collect cost information, calculate the rate of the cost drivers and estimate the learning factors for production activities.

Existing accounting data is analyzed and is allocated into activity costs according to the identified key activities and resources. The activities are specified and the total costs of each activity for the existing products are collected. The number of cost drivers for activities is retrieved. The rate of the cost driver for each activity is calculated by:

$$R = \frac{\text{Total cost on an activity}}{\text{Total quantity of cost driver}} \tag{21}$$

According to the concept of the learning effects, the tasks get faster with practice. Therefore, the cost driver consumption per production unit will decrease with a discount as the production volume increases. Take the setup activity as an example. In this case, the relationship between the average activity consumption and the production volume is shown in Figure 2. The average activity consumption for setup activity decreases as the production volume increases. This picture shows that the learning effect works for the setup activity. The average activity consumption, the number of cost drivers per production unit, needs to be specified to estimate the involved indirect costs. Similar to the traditional learning functions, the first unit activity consumption and learning factors are unknown and should be determined to construct the learning functions. Since some production activities take place in multiple components, and it can be reasonably assumed that an activity has the same learning rate in different product components. A

matrix methodology by Young et al. (2008) is adjusted to calculate each activity's learning rate. This method can obtain a learning factor of the same production activities involved in different parts by pooling activities in different parts into a single dataset. In this method, the learning factor and the theoretical first activity consumption are obtained with improved accuracy and increased confidence. It should be noted that different from the traditional learning effect functions where the first unit cost is calculated, the first unit activity consumption is obtained in the adjusted approach. A power learning function is assumed and shown in (22).

$$y_{i,j,k} = b_{i,j}x_{i,j,k}^{\beta_{i,}} \tag{22}$$

where $y_{i,j,k}$ is the average activity consumption for activity i in component j of product k, $b_{i,j}$ is the theoretical first unit activity consumption and $x_{i,j,k}$ is the production volume of component j in product k. This power function is converted into its linear format in Equation (23):

$$\ln y_{i,j,k} = \ln b_{i,j} + \beta_i \ln x_{i,j,k} \tag{23}$$

Figure 3.14: Learning effect in ABC
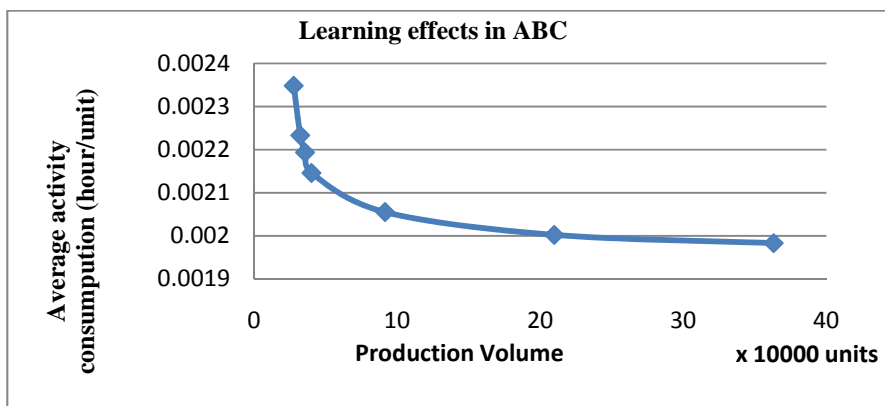
A linear matrix is presented

85

$$
\begin{bmatrix} \ln y_{i,1,1} \\ \ln y_{i,1,2} \\ \vdots \\ \ln y_{i,1,n} \\ \ln y_{i,2,1} \\ \vdots \\ \ln y_{i,2,n} \\ \vdots \\ \ln y_{i,J,1} \\ \vdots \\ \ln y_{i,J,n} \end{bmatrix} = \begin{bmatrix} \ln b_{i,1} \\ \ln b_{i,1} \\ \vdots \\ \ln b_{i,1} \\ \ln b_{i,2} \\ \vdots \\ \ln b_{i,2} \\ \vdots \\ \ln b_{i,n} \\ \vdots \\ \ln b_{i,n} \end{bmatrix} + \begin{bmatrix} \ln x_{i,1,1} \\ \ln x_{i,1,2} \\ \vdots \\ \ln x_{i,1,n} \\ \ln x_{i,2,1} \\ \vdots \\ \ln x_{i,2,n} \\ \vdots \\ \ln x_{i,J,1} \\ \vdots \\ \ln x_{i,J,n} \end{bmatrix} \beta_i = \begin{bmatrix} 1 & 0 & \cdots & \ln x_{i,1,1} \\ 1 & 0 & \cdots & \ln x_{i,1,2} \\ \vdots & \ddots & \cdots\cdots\cdots & \vdots \\ 1 & 0 & \cdots & \ln x_{i,1,n} \\ 1 & 1 & \cdots & \ln x_{i,2,1} \\ \vdots & \ddots & \cdots\cdots\cdots & \vdots \\ 0 & 1 & \cdots & \ln x_{i,2,n} \\ \vdots & \ddots & \cdots\cdots\cdots & \vdots \\ \vdots & \cdots & 1 & \ln x_{i,J,1} \\ \ddots & \cdots\cdots & \cdots\cdots & \vdots \\ \vdots & \cdots & 1 & \ln x_{i,J,n} \end{bmatrix} \begin{bmatrix} \ln b_{i,1} & \ln b_{i,2} & \cdots & \ln b_{i,n} & \beta_i \end{bmatrix} \qquad (24)
$$

Let

$$
Y_i = \begin{bmatrix} \ln y_{i,1,1} \\ \ln y_{i,1,2} \\ \vdots \\ \ln y_{i,1,n} \\ \ln y_{i,2,1} \\ \vdots \\ \ln y_{i,2,n} \\ \vdots \\ \ln y_{i,J,1} \\ \vdots \\ \ln y_{i,J,n} \end{bmatrix}, \qquad X_i = \begin{bmatrix} 1 & 0 & \cdots & \ln x_{i,1,1} \\ 1 & 0 & \cdots & \ln x_{i,1,2} \\ \vdots & \ddots & \cdots\cdots\cdots & \vdots \\ 1 & 0 & \cdots & \ln x_{i,1,n} \\ 1 & 1 & \cdots & \ln x_{i,2,1} \\ \vdots & \ddots & \cdots\cdots\cdots & \vdots \\ 0 & 1 & \cdots & \ln x_{i,2,n} \\ \vdots & \ddots & \cdots\cdots\cdots & \vdots \\ \vdots & \cdots & 1 & \ln x_{i,J,1} \\ \ddots & \cdots\cdots & \cdots\cdots & \vdots \\ \vdots & \cdots & 1 & \ln x_{i,J,n} \end{bmatrix}, \theta_i = \begin{bmatrix} \ln b_{i,1} & \ln b_{i,2} & \cdots & \ln b_{i,n} & \beta_i \end{bmatrix}
$$

then

$$
Y_i = X_i \theta_i \qquad (25)
$$

Solving Equation (25) through the least-squared-error approach, the parameters are determined by

$$
\theta_i = (X_i^T X_i)^{-1} X_i^T Y_i \qquad (26)
$$

3) Forecast the production volume and estimate the indirect cost for each product.

Forecasting the production volume in the family is a critical step in the adjusted ABC. The average activity consumption is estimated by substituting the production volume into Equation (22). For platform and non-platform components, the unit activity costs are calculated by:

$$
ic_{i,j,k} = R_{i,j} y_{i,j,k} \qquad (27)
$$

where $ic_{i,j,k}$ is the cost for activity i in component j of model k, $R_{i,j}$ is the activity rate for activity i in component j.

Since the production volume for a platform component is equal to the summation of the production volume in multiple products, its average activity consumption is significantly reduced due to centralized production and organization with improved production skills and experiences. The indirect cost of the individual products is obtained by aggregating the activity-related costs. In this way, the indirect costs are allocated to each product via the consumption of activity cost drivers.

3.4.2 Development Time

It has been claimed that the platform strategy will decrease production development time (Siddique, 2005). How much development time can be saved for implementing a product platform strategy needs to be addressed. Siddique (2005) put forward an Activity Based Time (ABT) estimation model to approximate the development time. The procedure to estimate the development time is similar with the procedure presented in Section 1.2, just substituting Cost with Time.

3.4.3 Customer Satisfaction

Meeting customer needs is the primary goal of any product. The criterion measures how well the customer needs are met by the platform (Otto and Holtta, 2005). Failure to meet customer's needs implies that the products supported by the platform will not sell to the known market segments that the products are currently targeting. The attributes include functional performance attributes as well as soft attributes (Weck et al., 2003). The functional performance attributes are defined as the functional attributes that can be

directly measured via physics based performance, such as the passenger volume of the automobiles, product weight, geometry and so forth. The soft attributes include the attributes that can impact product value, but are only measurable via customer surveys, such as styling and comfort of the automobiles.

The functional performance attributes are scored by converting utility values into 0-10 scale:

$$R_{ij} = 10U_{ij} \tag{28}$$

Where: R_{ij} is the score for attribute i in product j; U_{ij} is the utility values for attribute i in product j.

The utility values are calculated through the utility functions constructed in the same manner as presented in Section 3.3.2. The attributes that reflect customers' lifestyle, such as comfort, style, are scored by customers via a 0-10 scale evaluated by customers through customer surveys. Thus the performance attribute and lifestyle attributes are in the same commensurability.

The metric can be calculated by:

$$Y_{cs} = \sum_{variant\ j} \frac{1}{K_j} \sum_{Attribute\ i} w_{ij} R_{ij} \tag{29}$$

Where: Y_{cs} is the customer satisfaction for the whole family; K_j is the number of product attributes in product j; and w_{ij} is the weight of attribute i in product j.

It is believed that the higher the customer satisfaction score, the customer will be more satisfied.

3.4.4 Platform Extendibility

It is known that a product platform not only serves current products to fill needs in current market segments, but also serves the future market. Platform extendibility is a metric that reflects generational variety of product family, that is, the potential products the platforms can support. The more the potential combinations of platform elements, the more product variants the platforms can provide with less redesign efforts. For the multi-platform strategy developed in this research, the metric can be measured by the number of possible combinations of platform components.

Three types of platform relationships between platform components need to be considered to identify platform extendibility, including diversion feasibility, diversion simultaneity and energy capacity. Diversion feasibility determines whether the platform component can work appropriately with other platform components. Diversion simultaneity constraint implies one platform component must work with another simultaneously. Capacity constraint determines the energy balance and imposes an equation on the sum of input and output energy at respective components over a product. The platform extendibility is determined by the above three constraints.

In this chapter, the multi-platform strategy is described step by step. Three research problems mentioned in Chapter 1 are addressed with three research phases: Platform Identification Phase, Product Family Optimization Phase and Platform Alternative Evaluation Phase. In Chapter 4 and 5, a case study with seven (7) vehicles models is investigated to demonstrate the proposed method and the research results are studied to verify the multi-platform strategy.

Chapter 4 : Case Study

This chapter introduces background information about automobile industries and discusses the potential application of a platform strategy in multiple automobile models.

4.1 Introduction

A generic car is a wheeled motor vehicle used for transporting passengers, which also carries its own engine or motor. Automobiles are designed to run primarily on roads, to have seats for one to eight people, to typically have four wheels, and to be constructed principally for the transportation of people and goods. The primary function for automobiles is to transport certain numbers of passengers and amount of cargos comfortably, quickly and economically from one location to another one. In addition, automobiles must meet the government regulations and standards regarding emissions, fuel economy, and safety. The automobile design also involves the development of the appearance and ergonomics. The functional design and development of a modern motor vehicle is a complex task and is typically done by a large team from many different disciplines included in automotive engineers and industrial design. Everything from the engine to the tires has its own special universe of design and engineering.

The automobile is one of the most fascinating devices that a person can own. automobiles are also one of the most pervasive devices, with a typical American family owning two automobiles. There are various types of automobiles available in automobile markets. Different classification criteria will lead to different automobile classes. For example, based on an automobile's utility, it can be classified into sport car, middle-sized car, subcompact, city vehicle, etc. Based on its power sources, there are fuel, electric and

hybrid types of cars. These vehicle models favor different customer groups by providing different features. Though their targeted market segments are different, these vehicles do have some shared features and provide great opportunities motivating the platform-driven vehicle design and development.

Platform-based automobile development strategy has been applied by automobile manufacturers (Muffatto, 1999, Cusumano and Nobeoka, 1998, Bremmer, 1999, Bremmer 2000), such as K-series by Chrysler in the 1980s and J-platform by GM in 1978 and the underbody sharing across various brands in Volkswagen (Simpson, 2004). An automobile platform is a shared set of common design, engineering and production efforts, as well as major components of distinct models (Brylawski, 1999).

Vehicle platform-sharing combined with advanced and flexible-manufacturing technology enables automakers to sharply reduce product development and changeover times while a variety of vehicles are derived from one basic set of engineered components plus their distinct features (Schlie and Yip, 2000). In addition, the automobile platform strategy has become important in new product development and in the innovation process (Muffatto, 1999b). The finished products have to be responsive to market needs and to demonstrate distinctiveness while — at the same time — they must be developed and produced at low costs (Muffatto, 1999a). Adopting such a strategy affects the development process and also has an important impact on an automaker's organizational structure (Muffatto, 1999a). A platform strategy also offers advantages for the globalization process of automobile firms (Wihelm, 1997). The use of a platform strategy provides other benefits (Muffatto, 1999a), such as reduction of production complexity due to standardization, cost reduction through global resources deployment

and provision, mass production, etc. Other benefits from the platform strategy in automobile industries and module-based product family are discussed and summarized by Simpson (2004).

The platform strategy in automobile industries has been proven beneficial and advantageous to compete in the intense automobile market for automakers. Automobiles will be used in this research to illustrate the proposed multi-platform strategy.

4.2 Product Framework

The primary function for a vehicle is the transportation of people and cargo from location A to location B with safety and comfort. Three basic internal functional requirements of a vehicle have been identified to achieve the primary transporting function: propelling, housing, and towing (Weck, 2006). Propelling is the ability of the vehicle to roll on a surface as well as to accelerate and decelerate on command. The primary vehicle constituent parts responsible for this process are the power-train, the chassis and the wheels. The power-train consists of the fuel tank, engine, transmission, drive shaft and differential. The chassis is made up primarily of the structural underbody (carriage), the braking system as well as the suspension system. The wheels allow the vehicle to roll and transmit the torque generated by the engine to the road.

The second requirement, housing, is to provide room for the passengers and cargo, protecting them from wind, sun and other external elements. It also reduces drag and contributes to the external appearance of the vehicle. Customers usually use housing to characterize vehicle models. The towing capacity of a vehicle is primarily driven by the power of the engine and the ability of the chassis to transmit the towing load from the

hitch through the frame and onto the wheels. These three functional requirements exist in each vehicle model and are the common requirements for the vehicle family.

Corresponding to the three functional requirements, there are five engineering attributes (EA) to describe these three functions: passenger volume (PV), Cargo Volume (CV), towing capacity (TC), fuel economy (FE) and acceleration (AC). There are other EAs besides these five EAs, but these five are primary and widely accepted in various publications. Other attributes that can influence the values of vehicles and catch customers' interests, include mechanical quality, comfort, styling, reliability, interior features and service after sales. These attributes are definitely important for vehicles, but are not included in the consideration of functional engineering attributes. These attributes can be fully measured only after the vehicles are finalized and produced.

Four vehicle subsystems are considered to be related to these five EAs: power-train, chassis, body, and wheels. There are more components included in an automobile, but these four subsystems are common and representative across the vehicle family and consist of the major parts of vehicles. Thus, these four subsystems are regarded as the atomic units of vehicles in this research. Obviously, this automobile decomposition is not complete and can only serve as an illustration of the proposed method.

For each subsystem, there are corresponding design variables that characterize the parts and assemblies of the product system. The power-train is characterized by the fuel tank capacity (FC) and engine displacement (ED). The chassis is defined by the wheel track (WT), wheel base (WB) and ground clearance (GC). The body is associated with overall total length (LT) and height (HT) and wheels are characterized by tire width (TW) and diameter (TD). Obviously there are more design variables in a complex automobile

design process. For the purpose of clear illustration, these nine design variables are the only design variables that will be used since they characterize the major subsystems and the data for these variables is readily available, and are sufficient to illustrate the proposed method for a complex product.

An automobile framework is shown in Figure 4.1. The relationships among the functional requirements, EAs and subsystems are shown in Figure 4.2.

Figure 4.1: Automobile framework (Weck, 2006)
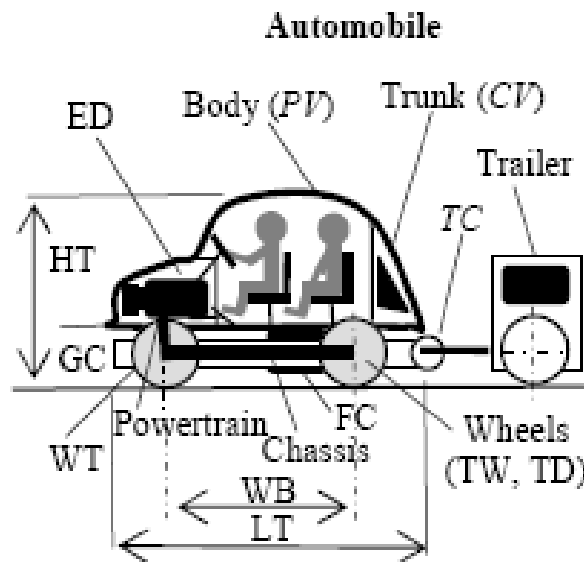
The research approach to support product family design and optimization is applied in a hypothetical automobile company, referred to as company X. While X sells an average vehicle and achieves an average sales volume in the market segment, X wants to explore the possible advantages of moving towards a platform strategy.

Figure 4.2: Automobile functional requirements, engineering attributes, subsystems and design variables
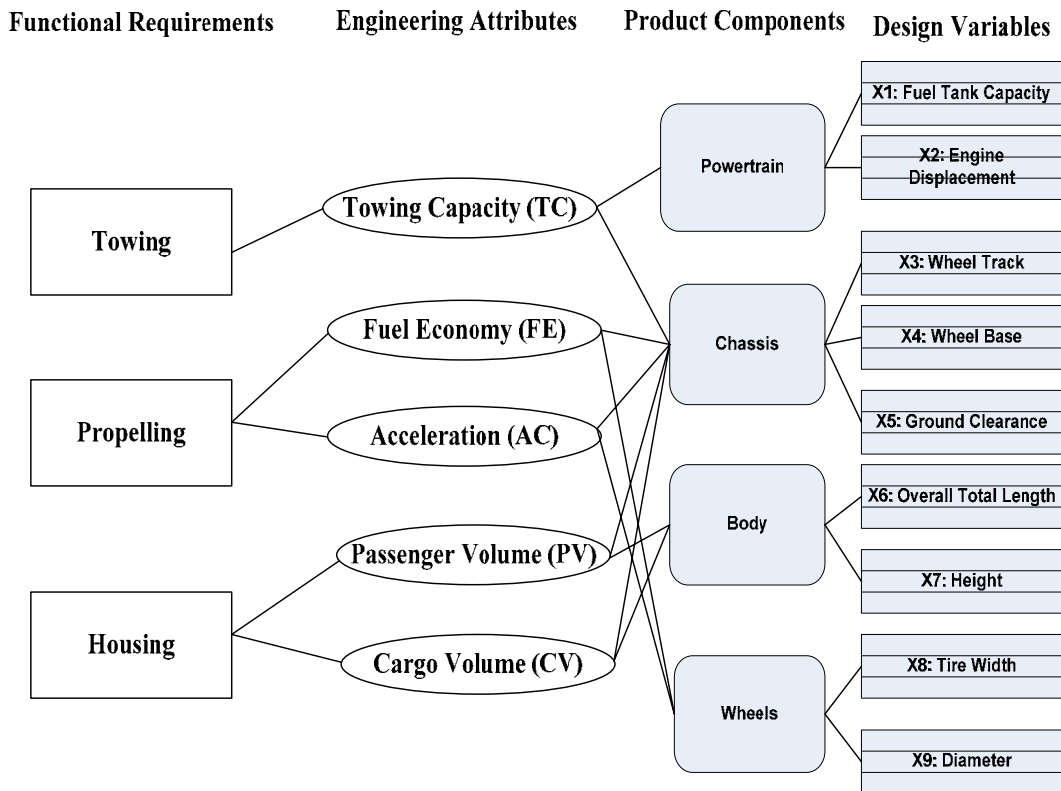
In this case study, 7 automobile models are analyzed to find out whether a platform-based product development strategy can be implemented in these 7 automobile models. These 7 automobiles have specific characteristics and are intended for different customer groups. They are focused on family utility and are designed to satisfy the demands from 7 market segments: small passenger sedan (SML), medium sedan (MED), large size sedan (LRG), sports car (SPT), minivan (VAN), sport utility vehicle (SUV) and light truck (TRK). While for each market segment, there are several types, such as S, LE, XLE and XRS, the configuration of these vehicles are almost the same and the differences between these vehicles are not large enough for a platform strategy. To illustrate the multi-platform strategy, it is assumed that for each market segment, X only produces one model. Other

types of models in the same market segment can be derived from the studied model by adjusting configurations. Thus the automobile family in this research will have 7 products and cover 7 market segments. Other vehicle types, such as a bus, which are mostly used in public transportation or other specific usages, are not included in the scope of this research.

SML is small in size and fuel-thrifty and needs little power to drive from one location to another. The buyers can find the lowest priced new automobiles in the small car class. MED is a relatively larger sedan that offers good safety and fuel consumption. MEDs are popular for U.S. consumers looking for somewhat larger vehicles that can be used for daily commuting, family errands, and vacations along with cargos carrying capacity in the trunk. LRG is known for its ability to carry up to six people and its ample trunk space. With bigger wheels, a sport-tuned suspension, larger brakes and lower rider height, Sports sedan (SPT) are to provide sports car enthusiasts with racing performance specification and fast driving feelings. VAN is taller than a sedan and designed for maximum interior room for personal use. SUV is usually equipped with four-wheel drive for on/off road ability. TRK is commonly used for carrying goods and materials with big carrying capacity. The automobile family with seven models is shown in Figure 4-3. Though each model in the family has its own alluring and differentiating characteristics, automobile manufacturers can implement platform-based strategy, such as sharing chassis among different models. The Chrysler K platform is an example of such a platform-based strategy.

Figure 4.3: Automobiles

SML MED LRG

SPT SUV

VAN TRK

To avoid the failure of loss of distinctiveness in the Chrysler K platform, a multi-platform strategy is implemented to show the identification of platform subsystems, the configuration of product variants and the determination of design specifications. For illustrative purposes, the automobile architecture is simplified. Only the primary value-delivering requirement (transporting) of the automobile is considered. This is regarded as the common customer requirements to the automobile product family. The detailed information about the existing 7 vehicle models can be found in Table B.1 and B.2.

Chapter 5 : Research Results and Summary

This chapter discusses the research approach through its application in an automobile case study. As introduced in Chapter 4, seven automobile models are analyzed to illustrate the implementation of the method.

5.1 Phase 1: Multi-platform Structure Identification Phase

5.1.1 Step 1: Value Analysis

Recall the common customer requirements for seven automobile models are propelling, housing, and towing. There are five principal functional engineering attributes (EAs) considered: passenger volume (PV), cargo volume (CV), towing capacity (TC), fuel economy (FE) and acceleration (AC). All of seven vehicle models have these five EAs. For each vehicle model, customers have different and specific needs while searching in the automobile market. Some want to get automobiles with larger space (PV and CV) for a big family, some focus on economic issues (i. e, FE), and some enjoy wild and speedy driving (acceleration). Though these seven vehicle models have a same set of EAs, need for diversity shows different importance when distinguishing between vehicle models. These importance values indirectly express different product values on different automobile models with regard to same EAs.

Market surveys are used to obtain the customers' opinions by asking questions such as "Which of the following features most attract attracts you to purchasing vehicle A?". Scales 1-9 are used to represent customers' agreement or disagree level: 1-totally disagree, 5-So-So and 9-totally agree. A sample survey can be found in Appendix B.3 and the relative attribute significance (RAS) is obtained for each vehicle model. Further,

normalized RAS is obtained and shown in Table 5.1, which captures customers' preferences to these seven vehicles.

Table 5.1: RASM for (7) seven automobile models

	PV	CV	TC	FE	AC
SML	0.21	0.14	0.17	0.31	0.17
MED	0.31	0.19	0.12	0.19	0.19
LRG	0.27	0.19	0.23	0.12	0.19
SPT	0.22	0.06	0.06	0.17	0.50
VAN	0.32	0.32	0.11	0.18	0.07
SUV	0.29	0.11	0.25	0.11	0.25
TRK	0.24	0.28	0.28	0.07	0.14

Values in cells indicate relative product importance of EA (column) in corresponding vehicle models (row). Consider SPT and VAN as examples. Cargo capacity and fuel economy are not typical considerations in SPT since sports cars are not designed for carrying much and energy savings. Thus, the values for CV and FE in SPT are very low (0.06) and (0.17). Also sports cars are not recommended for towing trailers or other heavy loads, thus, its towing capacity is not important (0.06). One of attractive points in SPT is fast driving to satisfy the needs of race enthusiasts. Acceleration time (0-100kph) is an important factor for fast-driving. This makes the AC get a considerably higher value of relative attribute significance (0.5). On the contrary, the case of van/minivans is a little different from sports cars. Van/minivans require large space for carrying passengers and cargo and have no specific requirements for their driving speeds. Therefore, van/minivans have higher values for PV (0.39), CV (0.26) but a relatively lower value for AC (0.09)

Next, a EAs and product subsystems correlation matrix is built on designers' experiences and expertise. The purpose is to map EA values into corresponding vehicle subsystems. In a way similar to the establishment of the RASM, a designer/engineer survey is designed to determine the relationship between EAs and four vehicle subsystems (e.g., from very weak to very strong relationships). RCM is derived from designer/engineer surveys and results are shown in Table 5.2.

The results in Table 5.2 show that PV and CV are closely related to vehicle body and chassis, but have no connection with power-train and wheels. The function of towing in different vehicle models is realized by the appropriately designed power-train and chassis. FE and AC have strong relationships with power-train and wheels. Though chassis can affect FE and AC, the effects are not so significant (0.11 and 0.06 respectively).

Table 5.2: RCM for the automobile family

	Power-train	Chassis	Body	Wheels
PV	0	0.22	0.78	0
CV	0	0.22	0.78	0
TC	0.82	0.18	0	0
FE	0.47	0.11	0	0.42
AC	0.44	0.06	0	0.5

The subsystem values in individual products are obtained by the multiplication of RASM and RCM, as described in Section 3.2.1. Then the subsystem values for the whole product family are calculated by summing up the subsystem values in individual products, shown in Table 5.3.

From Table 5.3, the vehicle chassis has the least values for the whole product family and vehicle body has the highest values. This means that making vehicle chassis as a platform subsystem will result in the least value loss of the whole family. On the contrary, the sharing of the vehicle body would cause the reduction of product differentiation inside the family and the lack of competition with outside competitors since the total value for vehicle body is the highest. If only subsystem values are considered to determine the product platform, chassis would be the ideal choice. Choosing vehicle chassis as a platform subsystem is a commonly applied strategy in current automotive manufacturing companies, such as the chassis sharing in the models of Chrysler brands.

Table 5.3: Subsystem values in individual products and the whole family

	Power-train	Chassis	Body	Wheels
SML	0.29	0.16	0.35	0.20
MED	0.27	0.16	0.39	0.18
LRG	0.24	0.17	0.44	0.15
SPT	0.35	0.12	0.22	0.32
VAN	0.21	0.19	0.51	0.10
SUV	0.37	0.16	0.31	0.17
TRK	0.32	0.18	0.40	0.10
TCV*	2.04	1.13	2.61	1.22

*Total Subsystem Value

5.1.2 Step 2: Redesign Effort Analysis

For each vehicle model, there are desired or targeted EA values and these desired values are shown in Table B.4. The desired EA values can be specified values by designers or derived from information of existing products in the markets, such as average values or the values of market leaders in the same market category. The design team knows exactly the relationship between subsystems and EAs, and then will estimate the redesign efforts

for each subsystem in each vehicle to reach the targeted EA values. The total redesign efforts of the family are obtained by summing up the redesign efforts of individual products. The estimated redesign efforts for the vehicle subsystems are shown in Table 5.4.

Table 5.4: Redesign efforts for automobile subsystems

	Power-train	Chassis	Body	Wheels
SML	1	1	1	1
MED	9	1	0	1
LRG	9	1	1	1
SPT	1	1	1	1
VAN	1	2	1	1
SUV	2	1	0	1
TRK	6	1	0	1
TRE*	29	8	4	7

Total Redesign Effort

The values presented in Table 5.4 show that great efforts will be spent while obtaining vehicles' desired EA values. This can be roughly explained as follows: the power-train is related to TC, FE and AC, and the existing products have relatively great gaps to the desired EA values (Table B.5) to their targeted values with regard to TC, FE and AC (i.e. 51% gap in TC for LRG, 85% in TC for VAN, 31% in FE for SPT, 21% in AC for LRG, 25% in AC for SUV). These gaps mean that great efforts are needed to modify power-train in each model to reach its targeted values.

Figure 5.1: Redesign Effort VS Total Values
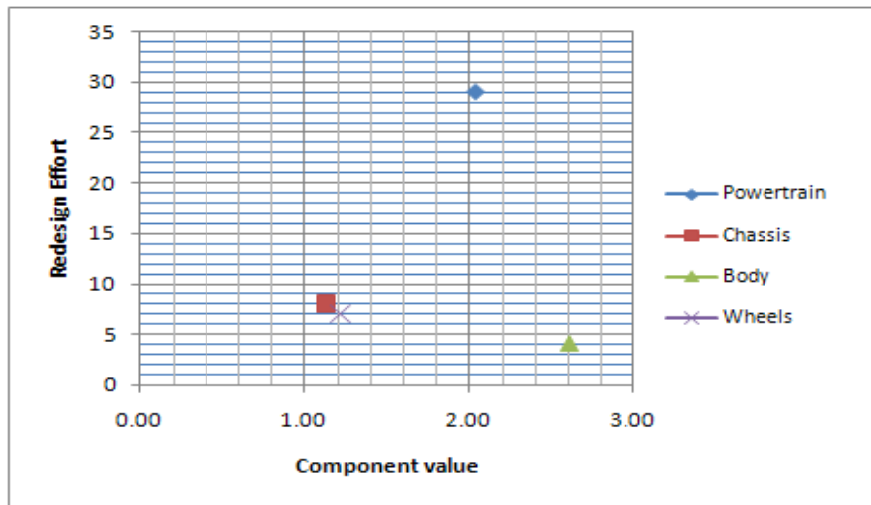
5.1.3 Step 3: Platform Identification Based on TCV and TRE

The subsystems that can be shared across these seven vehicles are determined by selecting the subsystems that require low redesign effort as well as low allocated product values. Though vehicle body requires low redesign effort, its product value is higher since the body is related to the passenger volume and the cargo volume, and carries great values to differentiate vehicle models from each other (some customers distinguish vehicles based on their body style). Thus the vehicle body is not recommended as a platform subsystem. Power-train has a high redesign effort value and subsystem value, thus it is not chosen as a platform subsystem. The number of shared subsystems is flexibly determined by changing the selection threshold values for the two criteria. From Figure 5.1, chassis and wheels are selected as the platform subsystems for further analysis. In the existing products, chassis and wheels in different vehicle models have the same set of design variables but with different values, and the chassis and wheels in various models are named platform subsystem instances.

5.1.4 Step 4: Platform-Variant Assignment

Once chassis and wheels are selected as the platform subsystems, the clustering approach is applied to determine the instance grouping and platform-product assignment with regard to these two platform subsystems. The goal of the clustering approach is to group the similar subsystem instances in different products into the same clusters to reduce the number of instances. Ward's method (Ward, 1963), an alternative agglomerative approach for hierarchical cluster analysis, is applied and in each step of Ward's method the minimal error sum of squares (ESS) or equivalently, the maximal r^2 is obtained. To eliminate the influence of scales on design variables, the values of design variables are standardized. The dissimilarity matrix for chassis and wheel are presented in Table C.1 and C.2 respectively. The ESS and r^2 are defined as follows:

$$ESS = \sum_i \sum_j \sum_k (X_{i,j,k} - \overline{x_{i,\cdot,k}})^2$$

$$TSS = \sum_i \sum_j \sum_k (X_{i,j,k} - \overline{x_{\cdot,\cdot,k}})^2$$

$$r^2 = \frac{TSS - ESS}{TSS}$$

where

$X_{i,j,k}$: the value for variable k in observation j belonging to cluster i;

$\overline{x_{\cdot,\cdot,\cdot}}$: the average value;

TSS: total sum of squares.

The cluster dendrograms for chassis and wheels are shown in Figure 5.2 and Figure 5.3 respectively. The agglomeration schedule for the chassis and wheels can be found in Table C.3 and C.4.

Figure 5.2: Dendrogram using Ward Method for Chassis

```
                    Rescaled Distance Cluster Combine

    C A S E        0         5        10        15        20        25
  Label     Num    +---------+---------+---------+---------+---------+

  MED        2      ─┐
  LRG        3      ─┴─┐
  SPT        4      ───┴─┐
  SML        1      ─┐   ├──────────────────────────────────────┐
  SUV        5      ─┴───┘                                       │
  VAN        6      ──────────────────────┐                      │
  TRK        7      ──────────────────────┴──────────────────────┘
```
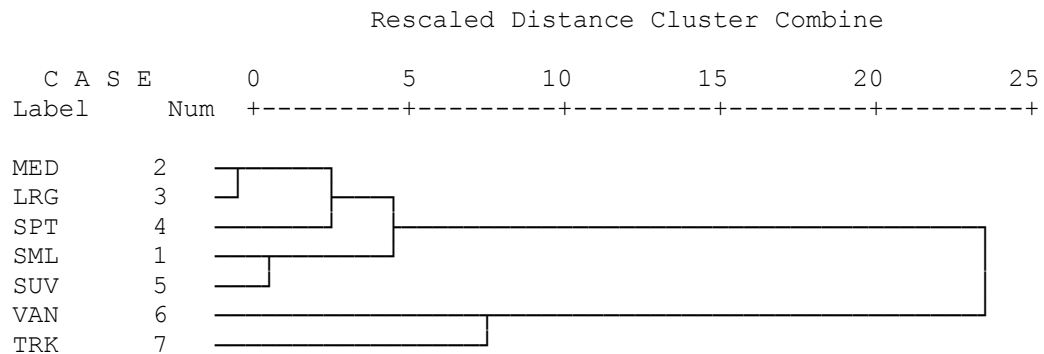
Figure 5.3: Dendrogram using Ward Method for Wheels

```
                    Rescaled Distance Cluster Combine

    C A S E        0         5        10        15        20        25
  Label     Num    +---------+---------+---------+---------+---------+

  SUV        5      ─┐
  VAN        6      ─┤
  MED        2      ─┼─┐
  LRG        3      ─┘ ├─────┐
  TRK        7      ───┘     ├───────────────────────────────────┐
  SML        1      ─────────┘                                   │
  SPT        4      ────────────────────────────────────────────┘
```
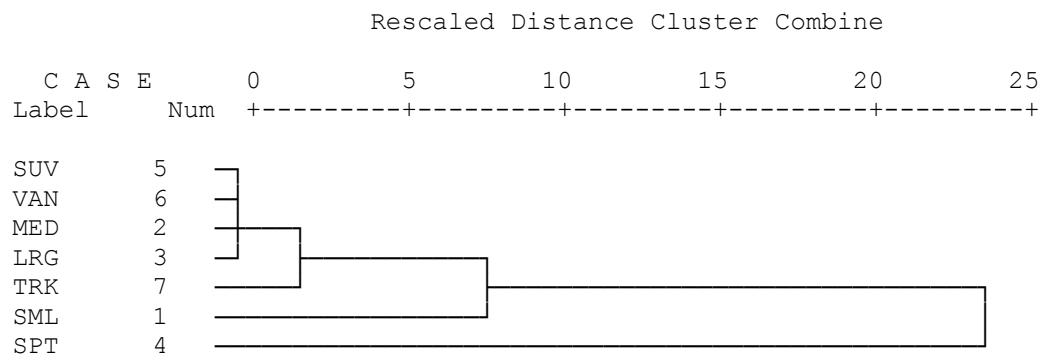
The advantage of a hierarchical clustering method is that the number of clusters can be flexibly determined by looking for gaps along the horizontal axis in Figure 5.2 and Figure 5.3. Take Figure 5.2 as an example. Starting from the right, there is a gap approximately from between 10 to 25, which suggests 2 clusters: one includes the chassis instances in SML, MED, LRG, SPT and SUV, and another includes VAN and TRK. There is another gap around 5, which suggests 4 clusters: cluster 1 with MED, LRG, SPT, cluster 2 with SML and SUV, cluster 3 with VAN and cluster 4 with TRK. The instances in one cluster

will be standardized and have same design values with regard to the corresponding subsystem. As a result, the number of platforms for one subsystem can range from 1 (all products have a same instance) to 7 (subsystems are independently designed), depending on instance dissimilarity. In this example, two clustering scenarios for chassis and two for wheels are considered, shown in Table 5.5 and 5.6**.**

Table 5.5: Product Platform Setting for Chassis

Chassis	Cha-1				Cha-2		
Cluster	1	2	3	4	1	2	3
Vehicle Models	MED, LRG, SPT	SML, SUV	VAN	TRK	SML, MED, LRG, SPT, SUV	VAN	TRK

Table 5.6: Product Platform Setting for Wheels

Wheels	Wh-1			Wh-2			
Cluster	1	2	3	1	2	3	4
Vehicle Models	MED, LRG, SUV, VAN, TRK	SPT	SML	MED, LRG, SUV, VAN	SML	SPT	TRK

Since the platform scenario is determined on the similarity of existing product subsystems, if the subsystem instance in a product variant is far from that in other product variant, this product variant will not necessarily share the subsystem with other product variant. For instance, cluster 3 and 4 in scenario Cha-1, only VAN is assigned to platform 3 and only TRK is assigned to platform 4. This indicates that the chassis in VAN and TRK have too many differences with other models, and are not appropriate to share a chassis with other vehicle models. Otherwise, forcing them to share a same set of design variables with others will make the VAN and TRK lose their uniqueness and power of competition.

The platform subsystem will be standardized and all product variants in the same cluster with regard to the subsystem will have same design specifications. For example, in Scenario Cha-1, SML and SUV share a chassis subsystem, thus the values for wheelbase and wheel track in SML and SUV are the same since wheelbase and wheel track are the design variables for the chassis subsystem. The values of design variables (platform and non-platform) are specified in the optimization step of Section 5.2.4.

There are totally 4 chassis and wheels combination settings, shown in Table 5.7.

Table 5.7: Chassis and wheels combination settings

	Setting 1	Setting 2	Setting 3	Setting 4
Chassis	Cha-1	Cha-1	Cha-2	Cha-2
Wheels	Wh-1	Wh-2	Wh-1	Wh-2

In Setting 2, 3 and 4, TRK is not sharing chassis and wheels with any other models since the differences between TRK and other models are too large to share. Under this situation, TRK can be individually designed. In the optimization phase, these four platform settings will be formulated in mathematical models and the design variables will be specified. The most promising platform setting which can bring the largest economic benefit will be selected as the final multi-platform strategy in the platform evaluation phase.

5.2 Phase 2: Product Family Optimization Phase

The purpose of mathematical optimization is to approach vehicle targeted engineering attributes subject to the design constraints as well as platform constraints. Mathematical models are formulated to specify the values of design variables in the four platform settings shown in Table 5.7.

5.2.1 Step 1: Identify Product Objectives and Construct Attribute Functions

Each vehicle model in the automobile family has its outstanding characters to cover its corresponding market segment, satisfy various customers' needs and distinguish between models. In short, each model has at least one critical EA that attracts the customers from one segment to another. For example, energy saving is a differentiating feature of a small sedan from a large sedan. Consequently, for a small sedan, the company wants to improve its FE to its extreme. In this research, for the purpose of demonstration, three objective EAs are formulated as objective EAs. The first one f_1 is the fuel economy (FE) of SML car. It is desired to reach its maximum value in this reasonable range. The second one f_2 is the passenger volume (PV) of a MED car since there is strong sensitivity of the medium segment to passenger volume relative to other engineering attributes. In addition, passenger volume is also one of the differentiating attributes that separate SML, MED and LRG, thus the company does not want MED's PV to overlay the range of SML and LRG sedans. It is desirable to stay close to its preferred value. The last objective EA f_3 is the cargo volume (CV) of VAN, which requires large space for family utilization. The CV in VAN is desired to reach its upper bound. In the automobile family, only these three objective EAs are used to illustrate the proposed methodology and the other primary EAs are modeled in the constraint set.

Table 5.8: Objectives for the automobile family

Objective	EA	Vehicle Model	Utility Type
f1	FE	SML	Largest is Best
f2	PV	MED	Value is Best
f3	CV	VAN	Largest is Best

In this step, the mathematical functions between EAs and design variables are presented in Appendix D. The information of products in each market segment is collected from a variety of sources such as the internet, magazines and textbooks. The raw vehicle data is presented in Appendix E. The mathematical functions for EAs are generated from data analysis about the existing products in the market with regression analysis. The parameters to construct the attribute functions are presented in Table D.1 to D.6.

5.2.2 Step 2: Utility Function Construction

In this step, the utility functions for the EAs are identified. The utility function types are determined by the design team with the consideration of product characteristics (external competition) and their differences from the products in the same family (internal competition). The objective function FE in SML is taken as an example to show the construction of its utility function. For SML, the goal is to obtain the FE values as large as possible to capture the eyes of customers in the SML market. With this approache, a Largest-is-Best type of utility function is applied. Also the range of FE for SML should be in the interval of SML market. In the historical data, the range of FE values in small sedan market (See Appendix E.1) is 19.5-41.5 mpg, thus f_{min}=19 mpg and f_{max} is 41.5 mpg. Therefore, the utility function for FE in SML is formed as:

$$u(FE_{SML}) = \begin{cases} 0, & f \leq 19.5 \\ \dfrac{f-19.5}{41.5-19.5}, & 19.5 \leq f \leq 41.5 \\ 0, & f > 41.5 \end{cases}$$

Other utility functions, including the objective functions and constraints, can be found in Appendix F. The objective EAs and the set of constraints (except the commonality constraints) are kept the same in the four scenarios for the purpose of result comparison.

5.2.3 Step 3: Obtain the Multiple Utility Objective Functions and Generate the Aggregated Objective Function (AOF)

As presented in Table 5.8, three objective utility functions are included in the mathematical optimization models and the corresponding utility functions are shown in Appendix F. The CAF^{2}M method described in Section 3.3.3 is applied in the three objective EAs to generate the aggregated objective function in the optimization model.

Table 5.9: 2^3 experiment design for the automobile family

Alternative	FE_{SML}	PV_{MED}	CV_{VAN}	Priority values				Average Priority value
				Run 1	*Run 2*	*Run 3*	*Run 4*	*P*
1	-	-	-	0.020	0.019	0.019	0.019	0.01925
2	+	-	-	0.041	0.032	0.030	0.029	0.033
3	-	+	-	0.059	0.054	0.051	0.049	0.05325
4	+	+	-	0.119	0.102	0.122	0.121	0.116
5	-	-	+	0.058	0.067	0.075	0.073	0.06825
6	+	-	+	0.136	0.142	0.129	0.132	0.13475
7	-	+	+	0.166	0.184	0.184	0.183	0.17925
8	+	+	+	0.400	0.410	0.389	0.394	0.39825

For each objective, two utility levels are selected: 0.3 and 0.7. Level 0.3 is indicated by '-' and level 0.7 id indicated by '+'. These two levels ensure the gap is big enough and at the same time not extreme values (0 or 1) in order to avoid confusion. In DOE phase,

since there are only three objectives, eight experiment alternatives (full factor 2^3) is generated, and the possible objective EA combinations are shown in Table 5.9.

To ensure an unbiased result, multiple engineers (four in this case study) are asked to evaluate the generated eight experiment alternatives and four pairwise-comparison matrices are generated with regard to the four engineers, corresponding to the 4 experimental runs. The sample of the pairwise-comparison matrix is shown in Appendix G (Table G.2 to G.5). Statistical software JMP is used to identify the significant factor effects and determine the significant 1st and 2nd order dependency. The significant factors include 1st order factor (FE, PV, and CV) and 2nd order interaction (FE*PV, FE*CV, and PV*CV). The parameters for CAF^2M and the consistency index are shown in Appendix G. The significant values from engineers are shown in Table G.6. The normalized parameters that are used to construct the aggregated objective function are show in Table 5.10.

Table 5.10: Parameters for the Aggregated Objective Function (AOF)

FE	PV	CV	FE*PV	FE*CV	PV*CV
0.17	0.24	0.27	0.10	0.10	0.12

Therefore the Aggregated Objective Function (AOF) is:

$$\text{Maximize AOF: } 0.17 * U_{FE_{SML}} + 0.24 * U_{CV_{MED}} + 0.27 * U_{CV_{VAN}} + 0.1 * U_{FE_{SML}}$$

$$* U_{PV_{MED}} + 0.10 * U_{FE_{SML}} * U_{CV_{VAN}} + 0.12 * U_{PV_{MED}} * U_{CV_{VAN}}$$

The constraints for the automobile functions include the upper and lower bounds of the design variables, as well as the commonality constraints. The commonality constraints indicate how the products share the design variables in the platform. When two products are derived from the same platform element, they will have the same values for the

corresponding design variables. For example, in platform scenario 1, SML and SUV have same chassis setting (Cha-1), thus the corresponding design variables of chassis (wheel track, wheel base and ground clearance) will own the same values in SML and SUV.

5.2.4 Step 4: Optimization

The mathematical optimization technique is applied to obtain the optimal values of design variables in the automobile product family. As mentioned above, the family objective is to maximize the value of aggregated objective function (AOF). The optimization problem is solved by Branch-and-Reduce Optimization Navigator (BARON), which facilitates the solution of non-convex and non-linear optimization problems to global optimality (Sahinidis, 2000). The four platform setting scenarios have the same objective utility functions and constraints except the platforming commonality constraints. The optimization results are shown in Appendix H. Table H.1 to H.4 present the optimization results for the platform scenarios. Not all objective EAs can reach their ideal values. For instance, in scenario 1, the PV in SML and CV in MED reach their ideal values (utility value=1), and the AC in SPT is close to the desirable level (0.991). However, the utility value of CV in VAN is relatively low. This can be explained that the three conflicting objective EAs compromise with each other to maximize the AOF value instead of optimizing individual objective EAs. The TC values for SML and SPT are blank in Table H.1 to H.8 since the towing capacity in most existing SML and SPT models is not recommended. Thus TCs for SML and SPT are not formulated in constraints to avoid the over-restricted feasible space.

Though the design specifications have been determined through the global optimization, this does not indicate the best design set has been determined. As known, the

112

determination of the final platform strategy is an iterative process. For the automobile models, only four major subsystems are discussed as a demonstration of the proposed method. The optimization only claims the best solution under the scope of discussion. The resulting optimal individual products are verified to determine whether they are consistent with the subsystem value loss analysis and redesign effort analysis. If they are not consistent, the platform subsystem determination process (in Phase I) should be restarted with adjustments.

The optimization model for the single-platform strategy, where all the automobile models have the same chassis and wheels, is formulated and solved. It is turned out there is no feasible solution for the single-platform strategy. It indicates that the single-platform strategy cannot offer an acceptable balance among these seven vehicles due to their great performance gaps. Forcing all vehicles to have same chassis and wheels is too demanding. The single-platform strategy does not work appropriately subject to the same set of constraints.

5.3   Phase 3: Platform Alternative Evaluation Phase

5.3.1 Engineering Economic Analysis

The automobile manufacturing costs are decomposed into three parts: direct cost, indirect cost and selling cost. The cost information about the vehicles is kept confidential and difficult to obtain, however, the media, such as internet, magazines and so on, provide a channel for us to know the product prices. Thus, the cost information about the existing vehicle models is derived from the product prices. It is assumed to be proportional to the selling price of the products. Based on the study in Vyas et al. (2000), it is assumed that

the average cost allocation proportion in automobile industries is 51% direct cost, 24% indirect cost and 25% selling cost. The selling cost of the product is proportional to the sum of direct and indirect costs. In addition, the average profit margin is around 9% of the total cost. Based on allocation percentages, the cost structure for the existing vehicle models is derived, shown in Table I.1.

The direct cost is related to production volume, thus the total direct costs are equal to the unit cost multiplied by the production volume. The direct costs allocated to each subsystem in vehicles are determined by the breakdown coefficient in Weck (2006). These coefficients indicate how the direct costs are deployed in vehicle subsystems in different vehicle models. The assumed cost breakdown coefficient is shown in Table I.2, which is adopted from Weck (2006). A subsystem has different cost percentages with regard to vehicle models, shown in Table I.2. According to these cost breakdown coefficients, the direct cost of vehicle subsystems in existing vehicle models are determined, displayed in Table I.3.

A costing-by-analogy method is applied to estimate the direct cost in the platform-based vehicles, which takes a known product and its direct cost as a reference and derives the cost of redesign subsystem by proportionally adjusting for changes in design variables. In this case study, the direct costs include direct material costs and direct labor costs and the existing product subsystems are taken as the reference baseline. It is assumed that the subsystem direct cost is correspondingly changed according to the subsystem's design variables, thus the unit direct cost of each redesigned subsystem is scaled with respect to the original design variable settings as follows:

$$c_{i,k}^{m'} = \frac{x_{i,1,k}^{'} * x_{i,2,k}^{'} \cdots x_{i,n_i,k}^{'}}{x_{i,1,k} * x_{i,2,k} \cdots x_{i,n_i,k}} c_{i,k}^{m}$$

Where:

$c_{i,j}^{m}$ is the original direct cost of subsystem i in model j;

$c_{i,j}^{m'}$ is the corresponding redesigned subsystem i in model j;

$x_{i,j,k}$ is the design variable k with regard to subsystem i in model j;

$x_{i,j,k}^{'}$ is the redesign variable k with regard to subsystem i in model j, which is obtained in the optimization phase;

n_i is the number of design variables for subsystem i.

For the platform subsystem that is shared across multiple automobile models, its direct cost is obtained by averaging the estimated direct costs in the corresponding models.

The indirect costs include production-related costs, such as R&D and engineering; business-related costs, such as corporate staff salaries and pensions; or retail-sales-related costs, such as dealer support and marketing (Vyas, 2000). The production-related overhead costs are the most important part among the indirect costs. These indirect costs are recovered by allocating them to each vehicle. The indirect costs are shared by the product family, and the adjusted ABC approach in Section 3.4.1 is implemented to estimate the indirect cost for the new platform-based product. For the purpose of illustration clarity and simplicity, only five major production activities (setup, machining, handling, maintaining, and supporting) that cause the presence of indirect costs are included in the adjusted ABC method.

The cost drivers for activities are shown in Table 5.11. These production activities take place in each product subsystem across the automobile family.

Table 5.11: Direct cost and indirect cost

Cost type	Activity	Cost driver
Direct cost	-	# of production volume
Indirect cost	Setup	Setup time(hours)
	Machining	Machine hours
	Handling	Production runs
	Maintaining	# of Labors
	Supporting	# of production volume

In the ABC model, the learning effect factors for the manufacturing activities are considered, except the selling cost which is assumed to be proportional to the total manufacturing costs (Vyas et al., 2000). A production activity takes place through different vehicle models, and the learning factor for an activity is assumed to be the same for different vehicle models. The sets of activity data for different vehicle subsystems are used in matrix form to estimate a single learning rate for the activity. The learning factors for activities are determined by a matrix-based method, proposed by Young et al. (2008). The learning factor for each activity is shown in Table 5.12. Based on these learning factors, the indirect costs in the platform-based vehicles are estimated.

Table 5.12: Learning factors for activities

Activity	Learning factors
Setup	0.94
Machining	0.85
Handling	0.85
Maintaining	0.92
Supporting	0.94

The estimated product manufacturing costs for each platform scenario are shown from Table I.6 to Table I.9. Note that not all models developed using platform strategies have lower estimated cost than their original costs (i.e., VAN in platform setting scenario 1, SPT, SUV, VAN, and TRK in platform setting scenario 2). The SML model in all scenarios has higher costs than its original costs. This can be explained by observing that the SML model is sharing subsystems with other models. Comparing it to the original design, the subsystems are overdesigned, resulting in the increase of direct and indirect costs. The results indicate that platform-based product development does not necessarily reduce the individual product's manufacturing costs, but rather the costs are reduced over the entire family. The cost saving of platform-based product development comes from the mass production of multiple products.

Platform based product development is beneficial in two ways: (i) it generates significant revenues, and (ii) the per-unit cost is lowered due to the effects of a learning curve. Revenue improvement requires the consideration of product price and sales. In order to compare different platforming scenarios, the capital investment costs (assumed only in the first year) as well as revenues from sales are taken into account to estimate the profit that the platform-based method brings to the manufacturer. The annual forecasted profit for the ith year and the jth scenario and the whole family can be estimated as:

$$A_{i,j} = \sum_{k=1}^{7} D_{i,k}(P_{k,j} - C_{k,j})$$

Where:

$A_{i,j}$ = the annual forecasted profit

117

$D_{i,k}$=the demand for the kth model

$C_{k,j}$=the cost for the kth model

$P_{k,j}$=the price for the kth model

The merit of a particular product realization program is assessed by its net present value (NPV). Net Present Value (NPV) with 12-year life cycle is obtained to assess the total manufacturing costs with the assumption of stable annual demands and discount rate (i.e., r=0.06). Product price is set to be equal to the product's original price based on the expectation that the redesigned products favor customers' taste just as the original products and do not change the values of the product. It is also assumed that the platform investment F for each platform is fixed and the same. The investment covers the plant setup, R&D and other costs that cannot or is difficult to allocate to production costs. The total investment is estimated by multiplying the unit investment for a platform with the number of the platforms.

The NPV is:

$$NPV_i = \sum_{i=0}^{N_p-1} \frac{A_{i,j}}{(1+r)^i} - n_j * F_j$$

where: n_j=the number of platforms

   F_j=the unit investment

The profit improvement of each platform strategy is calculated by

$$Cs_i=NPV_i-NPV_0$$

where:

118

$\mathrm{Cs_i}$=the cost saving for the ith platform strategy

$$NPV_o = \text{the original NPV} = \sum_{i=0}^{N_p-1} \frac{A_o}{(1+r)^i}$$

The NPV for each platform strategy is shown in Figure 5.4:

Figure 5.4: NPV for each platform scenario with unequal demands
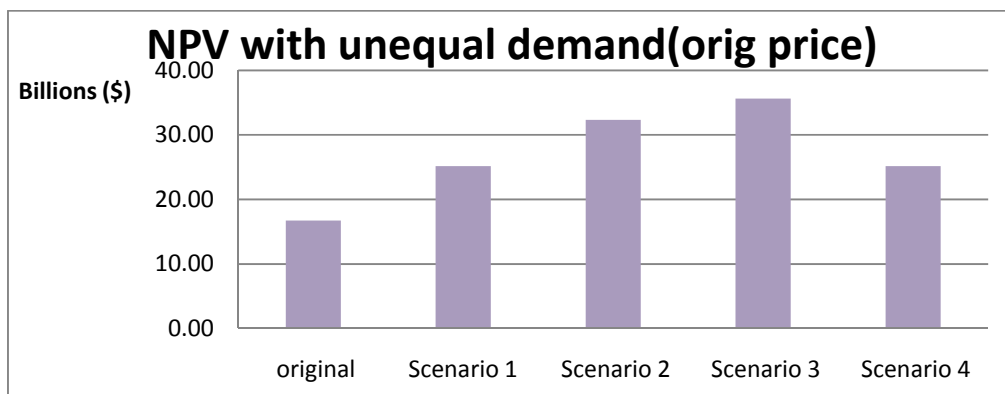
In Figure 5.4, NPV is calculated under the assumption that the prices are the same as their original prices. Thus, platform setting 3 has the largest NPV. The profit of the whole family improves about $20 Billion.

To analyze the influence of pricing strategy and product demand, another three cases are studied: equal demands with the new prices, unequal demands with new prices, and equal demands with original price. The new prices are hypothetically set to be proportional to product costs with a margin profit ρ. The results for these three cases are shown in Figure 5.5, 5.6, and 5.7.

Figure 5.5: NPV for each platform scenario with equal demands

NPV with Equal demand(new price)

Billions

original — Scenario 1 — Scenario 2 — Scenario 3 — Scenario 4
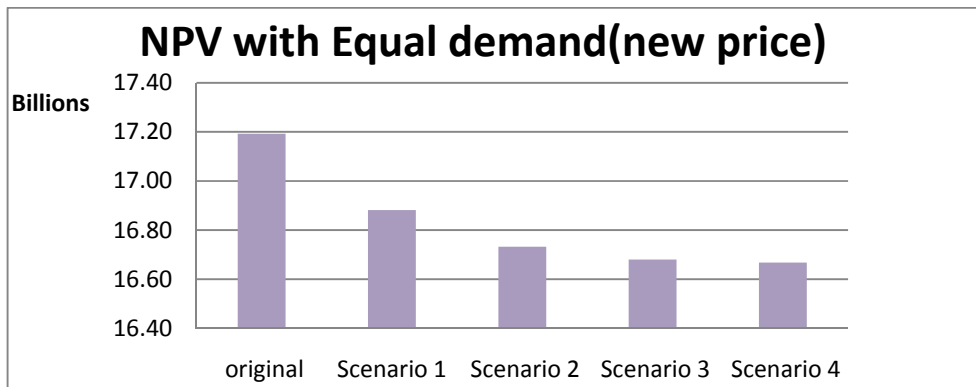
Figure 5.6: NPV for each platform scenario with equal demands

NPV with unequal demand(new price)

Billions

original — Scenario 1 — Scenario 2 — Scenario 3 — Scenario 4
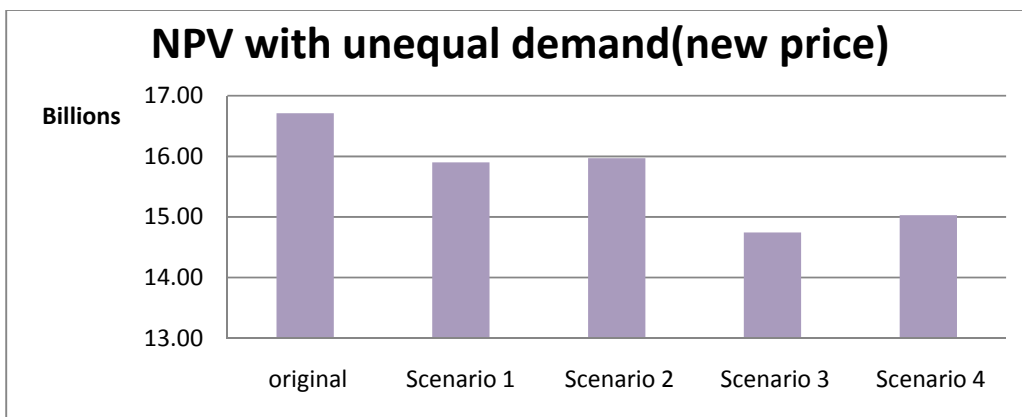
Figure 5.7: NPV for each platform scenario with equal demands

NPV with equal demand(original price)

Billions

original — Scenario 1 — Scenario 2 — Scenario 3 — Scenario 4
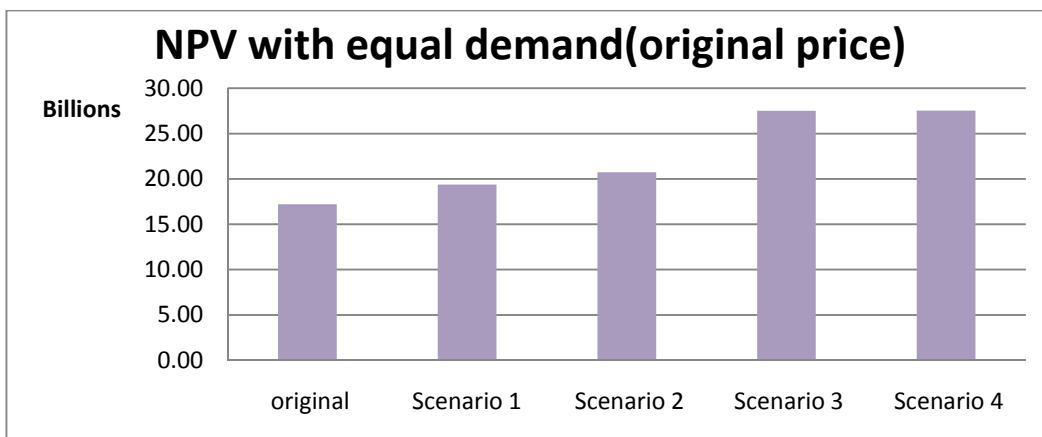
As shown in Figure 5.5 and 5.6, if a new price strategy is implemented based on the product cost, the original NPV is higher than all the platform scenarios. This result is

explained by recognizing that the new prices are proportional to the product costs, and the increase or decrease of the product costs will result accordingly in the corresponding change of margin profits. As shown in the cost structure Tables I.6, I.7, I.8 and I.9, in most scenarios, most of the unit costs are lowered due to the mass production and management; consequently, the profit margin decreases. This indicates that the platform strategy does not bring any benefits to the company, since the pricing strategy will influence decision making concerning product family development.

If the production volume is changed to equal demands, platform Setting 4 would be the most beneficial one instead of Scenario 3. The results indirectly validate that the product demand distribution impact the number of platforms, as shown Seepersad et al. (2000).

5.3.2 *Other Issues Related to Automobile Case Study*

While the product platform strategy and multi-platform method present an end-to-end view of product family design and development, it should also be recognized that a significant number of simplifying assumptions have been made. These assumptions are necessary to simplify the demonstration of the proposed method. There are some other factors that need to be considered to obtain more general design results, including:

(1)   Product complexity. Four subsystems are abstracted from the automobile models. The real automobiles include much more subsystems except what have been discussed. In addition, the interrelationship between subsystems is ignored to simplify the vehicle models. Interdependencies will certainly exist and will complicate the product design model.

(2)   Soft attributes: The focus of the method is on product functional attributes, i.e., PV, CV, etc. Soft attributes, such as styling and interiors also play significant roles in product valuing and customers perceived value. The inclusion of soft attributes will require more compromises of functional attributes to achieve soft attributes' maximum values. It is assumed that the soft attributes have been built-in to the design of product subsystems.

(3)   Pricing strategy: The discussion above shows that pricing significantly impacts the results of platform strategy. It is assumed that product prices are set proportional to manufacturing costs. But in reality, prices are adjusted based on seasonal factors, competitors' action or inventory levels. For example, the price of a car decreases at the end of the year to stimulate the annual sales.

(4)   Platform extendibility: The service of platforms benefits multiple product generations. They are also expected to be leveraged for as many products as possible. The platform coverage is also a critical issue to consider.

Chapter 6 : Conclusion and Future Work

This research has led to the development of a method to determine the multi-platform strategy associated product variants and the design specifications. It has been demonstrated for seven (7) automobile products. This chapter reviews the research objectives and proposed research approach, including the highlights to answers of the research questions presented in Chapter 1. The resulting contributions and limitations of the research are then summarized. The closing remarks are provided and future research opportunities are discussed.

6.1 Research Summary

The primary purpose of this research has been to develop a generalized method that covers the issues related to platform planning, including the determination of platform components, the identification of the optimal platform setting for multiple products, and the configuration of the optimal product variants. For this purpose, design management tools, such as QFD and DSM are used to represent critical design information and facilitate the information translation among different design perspectives. In addition, a clustering approach is adopted to define platform settings and product-to-platform assignments. Furthermore, the product positioning problem is solved in the context of multiple criteria and the product specifications are determined in mathematical optimization models with the aid of utility functions.

The secondary purpose is to provide a cost estimation model to evaluate the platform-based product development and investigate the cost savings as the result of implementing the multi-platform strategy. For this purpose, a cost model is formulated and an adjusted

ABC approach is used to allocate the shared indirect costs into individual products and estimate the cost savings as a result of mass production of shared parts/components/features.

In summary, the proposed method provides an intuitive way that facilitates the formation of a multi-platform strategy to reduce the overleveraging of a single platform and avoid excessive performance compromises of individual products. In addition, a systematic method is presented to cover the issues from market through design specifications in the platform planning. The proposed method is verified with an automobile case study comprised of seven (7) vehicle models from seven (7) market segments. The automobile case study shows that the cost savings from the platform-based strategy is not necessarily apparent for each product, but for the entire family due to gains in economies of scale.

In particular, the application of this method to support multi-platform strategies and product specifications is introduced and exploited to address the following motivating research problems and sub-problems.

Problem 1: Which parts/components/systems should serve as platform components in the multi-platform strategy?

The platform assets are determined by steps 1-3 of the first phase in the proposed research method. The product asset values analysis and redesign effort analysis define which element/s can be shared among multiple products. The QFDs are used to collect common customer requirements, map them to product design attributes and separate the product values into product physical assets. DSM is adopted to represent the

dependencies and interrelationships between these physical assets. The platform elements are those with less value-added and less generational design efforts.

> Problem 2: Given a set of product variants, what is the optimal number of platforms to derive them from?

The number of platforms is flexibly determined through the clustering approach in step 3 of phase 1. The clustering approach is conducted based on product similarity. The products in the family are divided into several groups with regard to the platform assets. The number of groups is decided by the design team according to the desired commonality and distinctiveness level. When more than one physical product asset can be shared, the combination of the platform asset groups forms the multiple platforms. In addition, to make a comprehensive analysis, more than one platform scenario is formed and the platform scenarios are compared based on the metrics presented in phase 3 of the proposed method.

> Problem 3: How to configure product members with platform elements to address the targeted market segments and competitors?

This problem includes the product-to-platform assignment problem and product variety optimization problem. The product-to-platform assignment in the multi-platform strategy is addressed in the clustering approach in step 4 of Phase 1 while determining the product clusters. This assignment indicates which product is derived from which platform. The product specifications are determined in the optimization phase with product performance

attributes as objectives. Meanwhile, the product positioning issues are considered in the optimization formulation with help of utility functions.

> Sub-Problem 1: When addressing the three questions, how is the Voice of Customers incorporated into consideration and how are the values of individual products expressed and deployed in the product platform planning?
>
> Sub-Problem 2: For designers or engineers, how are their preferences of the product components concisely expressed in the substantiation of platform and product

The voice of customers is included in multiple means in the proposed method. First, the voice of customers is collected from the market and common customer requirements for multiple products are extracted. Second, the product values are allocated into product component, parts or other physical elements based on customers' evaluation. To reflect designers' or engineers' design preferences, various types of utility functions are built and formulated into the optimization model. In addition, CAF^2M takes the relative significance of multiple engineering attributes into account to address product competition and allocation issues.

> Sub-Problem 3: Given several platform alternatives, what metrics can be used to assess, identify and determine the most beneficial and promising one?

Several metrics are presented to assess the multi-platform strategy. These metrics are presented in Section 3.4 and include product cost, development time, customer satisfaction and platform extendibility. Due to the limitation of data availability, only

economic analysis is conducted in the case study to show how the mass production can bring economic benefits to companies. Other metrics remain on the methodology level.

In summary, the proposed method answers the motivating research problems presented. To show how to implement the method, an automobile case study is developed.

6.2 Contributions and Limitations of the Research

This research has contributed in facilitating the implementation of a multi-platform strategy in the redesign of existing products. The proposed method provides a manner that can help to determine the common elements across the products and define the sharing of these common elements among multiple products. The multiple products can be effectively derived from the multiple platforms.

One of the contributions lies in the determination of the sharable assets (component, parts or features) among multiple products and the definition of platform structure in an intuitive and flexible way. A two-phase QFD approach is used to capture design information and hierarchically deploy product values into product assemblies. A clustering approach was proven to be a practical, flexible means to divide products into smaller groups based on their similarity. Meanwhile, the product-to-platform assignment is solved in the clustering procedure.

Another contribution in this research is to apply the concept of multi-objective optimization model along with utility functions to address the product positioning and competition problem. The multi-objective optimization model is to optimize product variety subject to a set of commonality constraints. The utility functions, which can quantitatively reflect decision makers' preferences, are combined in the optimization

models to make the multiple objectives comparable. In addition, a method that deals with multiple attributes and addresses their interdependencies with consistency checking is put forward and adjusted to convert a multi-dimension optimization problem into a single-dimension one.

To assess the potential economic benefits, especially for the indirect costs that are lumped together in the product family development, an adjusted Activity-based Cost (ABC) estimation technique is presented to estimate cost savings as a result of the multi-platform strategy on investigating the available accounting system. This adjusted ABC takes the economy of scale into account and helps to allocate the shared indirect costs into individual products with the inclusion of learning effects. The ABC model with learning effects clearly expresses how platform-based product development reduces the production costs by sharing.

6.3 Future Work

The proposed research emphasizes the implementation of a bottom-up strategy to redesign a set of existing products. Different than the single-platform strategy, a larger product family size can be supported by the multi-platform strategy. It is anticipated that the multi-platform strategy would help to reduce the performance compromise while sharing some common product features and satisfying the variety needs. The method is conducted based on the information of existing products, i.e., the clustering analysis is on the similarity of the existing products. Therefore, the method is not applicable for new product development if there are no modifications of the proposed method. Modifications to consider new product development are an area for future research.

In this dissertation, the automobile example developed by de Weck (2006) has been used as an illustrative case study to explain the proposed multi-platform strategy. This method systematically studies the automobile family from a different perspective, i.e., the application of bottom-up strategy to redesign the existing vehicle models and the concept of multi-platform strategy. Comparisons to de Weck's results with the new method can only be performed at a theoretical level at this point. For instance, the multi-platform strategy can reduce the performance loss of individual products since the platform is only shared by a subset of the product family. Comparisons on characteristics such as costs, customer satisfaction, etc. are not feasible without a real case study supported with data from a manufacturer. Future work will seek to conduct such a study to verify the advantages of the multi-platform strategy over single platform strategies.

The attributes of a product not only include the measurable performance attributes, such as the towing capacity and cargo volume, but also other soft attributes, such as appearance, styling and aesthetics. These soft attributes are difficult to quantitatively measure, but can be critical factors that attract customers' interest, especially when the measurable attributes are not visible to the customers or of limited information. In this research, the soft attributes are not fully considered; however, they cannot and should not be ignored in the design of platforms and product variants. Putting these soft attributes in the platform planning, product specifications and evaluation would be a great compliment to the research method of this dissertation.

As known, the products are competing with each other, even products that are from the same company. In this research, utility functions are used to represent the product positioning problems based on the information of existing products in the market. That is

not enough. The positioning of individual products and reduction of competition (interior and exterior) are complex issues and need more research. It is not as simple as avoiding performance overlaying. It involves the market analysis (static and dynamic), competitor analysis (existing and newly introduced), and other market issues. More research efforts are required to take care of product positioning problems.

It is known that product platforms have lifetimes that exceed the lifetime of the product variants that are derived from them. The generational service of product platforms requires the consideration of future market uncertainties, such as the changing customer needs and requirements, new technologies, new regulations or other potential changes. These uncertainties require platforms to be designed with flexibility to appropriately respond to the future. In addition, even the multi-platform based product family can provide more flexibility by offering more platform options than the single-platform strategy, fast developing technologies and intensive market competition are and will be continuously challenging the bandwidth of these platforms. This calls for methods for designing flexible product platforms to account for the expected and unexpected changes during the lifetime of the multiple platforms.

Finally, another possible research area in the future would be the creation of the ABC related accounting system. ABC is a great approach for cost estimation and evaluation for platform-based product development, but the lack of the corresponding accounting system causes the application of ABC to remain on a theoretical level. In this research, the cost estimation is conducted with an assumption that the products would be produced with a cost structure similar to the current production process. This assumption allows us to apply the adjusted ABC to estimate the costs related to product family. However, the

adjusted ABC method will yield significant estimation errors for innovative product designs with new production technologies. Therefore, it is an urgent and challenging task to design a cost system to apply the proposed ABC model for new technologies.

Appendix A: Modular Drivers and Commonality Index

Table A.1: Modular drivers (Adapted from Kreng and Lee, 2004))

Modular Drivers	Definition
Carryover	A part or a subsystem of system of a product not exposed to any design changes during the life of the product platform
Technology Evolution	Changes as a result of changing customer demands and technology shift
Planned Product Changes	Parts intend to develop and change to fulfill certain customer demands better, or decrease production costs
Standardization of Common Modules	Parts as possible candidates for common unit modules and can be produced in large production volumes.
Product Variety	The range of product models produced within a particular time to meet the various market demands.
Customization	Product in response to specific customers requirements by providing different modules.
Flexibility in Use	Rearrange or add some modules to achieve the desired and/or additional functions without drastically changing the whole product infrastructure.
Product Development Management	Concurrent engineering tends to rearrange design task into correlated subs-tasks by identifying interfaces and relationships among modules to substantially reduce developing lead-time.
Styling	Typically visible parts of the product to represent product identity
Purchasing Modularity Components	It is the vendors who take the responsibilities of manufacturing, development, and quality; therefore it is possible for customer to obtain high purchasing flexibility and better service by Providing standard modules instead of individual parts.
Manufacturability Refinement and Quality Assurance	Producing modules with similar manufacturing processes to reduce production cost effectively, utilize machines efficiently, and also assure quality easily.
Quick Service and Maintenance	Modularity can provide better maintenance and quick services.
Product Upgrading	Upgrade products by providing improved modules to meet future market requirements.
Recycling, Reuse, and Disposal	Provide environment-friendly products with "green" technology.

Table A.2: Commonality index (Thevenot and Simpson, 2004)

Name	Developer	Commonality	measure for Range	Focus
Degree of Commonality index (DCI)	Collier (1981)	Whole family	1-β	Number of common components
Total constant Commonality index (TCCI)	Wacker and Treleven (1986)	Whole family	0-1	Number of common components
Product line commonality index (PCI)	Kota et al.(2000)	Whole family	0-100	Number of common components
Percent commonality index (%C)	Rosen et al.(1998)	Individual products	0-100	Non-differentiating components
Commonality index (CI)	Martin and Ishii (1996)	Whole family	0-1	Number of common components, connections, and assemblies
Component part commonality CI(C)	Jiao and Tseng (2000)	Whole family	1-α	Cost of components

Appendix B: Automobile Information

Table B.1: Existing products in X

Market Segment	PV (cft)	CV (cft)	TC (pounds)	FE(miles/gal)	AC (seconds)
SML	92.0	16.6	1500	30.0	9.68
MED	101.4	15.0	1000	27.0	9.11
LRG	106.9	14.4	1000	23.5	6.09
SPT	73	10.2	NR	16	5.76
SUV	108.2	36.4	1500	25	8.74
VAN	177.4	140.6	3500	20	7.50
TRK	89	39.7	3500	22	10.51

Table B.2: Design specifications for vehicle models in X

	FC	ED	WT	WB	GC	HT	LT	TW	TD
	[g]	[ccm]	[in]	[in]	[in]	[in]	[in]	[mm]	[in]
SML	11.00	1200	60.0	109.8	6.0	58.0	174.0	175.0	15.0
MED	13.00	2000	61.6	109.5	5.5	57.8	180.0	215.0	16.0
LRG	20.00	3000	62.9	115.4	5.3	58.0	200.8	205.0	16.0
SPT	21.00	3218	60.4	102.1	4.9	53.4	174.2	231.2	17.0
VAN	20.00	3700	62.1	107.1	7.9	69.0	186.9	225.0	16.0
SUV	19.98	3899	65.2	117.9	6.6	70.4	197.6	225.0	16.0
TRK	20.61	3372	60.2	124.1	8.0	67.6	205.7	215.0	15.5

Table B.3 Sample survey results for EA evaluation

Which of the following features most attract attracts you to purchasing vehicle A? Please specify one value for each cell to indicate your option					
Please pick a number between 1-9 (1-totally disagree, 9- total agree)					
	Passenger volume	Cargo Volume	Towing Capacity	Fuel Economy	Acceleration Time
SML	2	2	5	9	6
MED	5	5	7	4	5
LRG	7	7	6	5	5
SPT	1	2	8	3	9
VAN	9	9	7	4	5
SUV	3	3	7	3	7
TRK	9	8	7	2	4

Table B.4: Target EA values for each market segment

EA		PV	CV	TC	FE*	AC
SML	Targeted	91.46	14.22	1472.00	27.73	9.06
	Min	72.9	10.5	750	19.5	5.57
	Max	97.9	20.6	2000	41.5	12.93
MED	Targeted	100.04	15.08	1783.97	24.97	8.07
	Min	88.2	10.6	1000	19	5.71
	Max	110.5	21.6	4000	49.5	14.45
LRG	Targeted	106.05	17.07	1506.45	21.27	7.35
	Min	102.3	14.4	1000	19	5.75
	Max	108	20.6	3417	23.5	9.74
SPT	Targeted	73.99	11.22	1768.00	21.00	5.87
	Min	41	3.7	1000	14	3.38
	Max	99	19	3527	31	9.77
VAN	Targeted	104.45	29.90	2774.19	21.42	8.72
	Min	90.1	120.1	1000	17.5	7.06
	Max	154	147.4	6000	27	10.867
SUV	Targeted	169.99	138.18	2970.00	20.15	9.35
	Min	160.5	17.2	1600	19.5	7.50
	Max	182	65.9	3500	23.5	11.53
TRK	Targeted	81.40	40.39	4980.91	19.05	8.83
	Min	52.5	33.5	2900	16	7.58
	Max	106.1	45.8	7100	22	10.51

Table B.5: EA gaps

EA gap	PV	CV	TC	FE	AC
SML	0.01	0.14	0.02	0.08	0.06
MED	0.01	0.01	0.78	0.08	0.11
LRG	0.01	0.19	0.51	0.09	0.21
SPT	0.01	0.10	0.00	0.31	0.02
VAN	0.03	0.18	0.85	0.14	0.00
SUV	0.04	0.02	0.15	0.01	0.25
TRK	0.09	0.02	0.42	0.13	0.16

Appendix C: Clustering Analysis Results

Table C.1: Dissimilarity matrix for chassis

Market Segment	Squared Euclidean Distance						
	SML	MED	LRG	SPT	VAN	SUV	TRK
SML	0.000	0.121	0.427	0.254	0.554	1.173	0.840
MED	0.121	0.000	0.139	0.204	0.621	0.751	1.163
LRG	0.427	0.139	0.000	0.613	0.869	0.384	1.185
SPT	0.254	0.204	0.613	0.000	1.095	1.669	2.001
VAN	0.554	0.621	0.869	1.095	0.000	0.772	0.732
SUV	1.173	0.751	0.384	1.669	0.772	0.000	1.208
TRK	0.840	1.163	1.185	2.001	0.732	1.208	0.000

Table C.2: Dissimilarity matrix for wheels

Market Segment	Squared Euclidean Distance						
	SML	MED	LRG	SPT	VAN	SUV	TRK
SML	0.000	0.361	0.361	2.000	0.361	0.361	0.250
MED	0.361	0.000	0.000	0.694	0.000	0.000	0.111
LRG	0.361	0.000	0.000	0.694	0.000	0.000	0.111
SPT	2.000	0.694	0.694	0.000	0.694	0.694	1.250
VAN	0.361	0.000	0.000	0.694	0.000	0.000	0.111
SUV	0.361	0.000	0.000	0.694	0.000	0.000	0.111
TRK	0.250	0.111	0.111	1.250	0.111	0.111	0.000

Table C.3: Agglomeration schedule for chassis*

	Cluster Combined			Stage Cluster First Appears		
Stage	Cluster 1	Cluster 2	Coefficients	Cluster 1	Cluster 2	Next Stage
1	2	3	.004	0	0	3
2	1	5	.099	0	0	4
3	2	4	.274	1	0	4
4	1	2	.536	2	3	6
5	6	7	.972	0	0	6
6	1	6	2.179	4	5	0

Table C.4: Agglomeration schedule for wheels*

	Cluster Combined			Stage Cluster First Appears		
Stage	Cluster 1	Cluster 2	Coefficients	Cluster 1	Cluster 2	Next Stage
1	5	6	.000	0	0	2
2	2	5	.000	0	1	3
3	2	3	.000	2	0	4
4	2	7	.089	3	0	5
5	1	2	.356	0	4	6
6	1	4	1.167	5	0	0

The agglomeration schedule is a numerical summary of the cluster solution. For Example, at the first stage, cases 2 and 3 are combined because they have the smallest distance and the cluster created by their joining next appear in stage 3. In stage 3, the cluster created in stage 1 and case 4 are joined. The resulting cluster next appears in stage 4.

Appendix D: Performance Functions

The mathematical functions between EAs and design variables are derived based on the suggestion of Weck (2006). The functions are introduced according to physical insight and response surface model. The coefficients in these functions are derived from least squared error (LSE) model. The functions are developed for EA at a time. For different vehicle models, the function structure are same, but with adjusted coefficients.

Passenger Volume (PV): The amount of available space in the passenger compartment is expected to scale with wheel track (width), wheel base (distance between axles), height and total length.

$$PV = p_0 + p_1.x_3 + p_2.x_4 + p_3.x_6 + p_4.x_7 + p_5.x_3.x_4.x_6$$

Cargo Volume: The cargo volume in medium sedans in primarily determined by the trunk space, which in turn depends on the width of the vehicle and the rear overhang (distance from rear axle to rear bumper).

$$CV = c_0 + c_1.x_3 + c_2.x_4 + c_3.x_6 + c_4.x_7 + c_5.x_3.x_4.x_6$$

Towing Capacity: In order to provide a large towing capacity a vehicle needs to have a strong engine (high horsepower rating), a sturdy chassis to sustain the axial loads induced by the trailer, a long wheelbase for directional stability, and a reasonably large curb weight relative to the load. Also, more subtly, frontal area (approximated as width times height) and ground clearance (center of gravity location) are relevant factors. We first approximate the horsepower rating [hp] of the vehicle as a function of engine displacement:

$$hp = h_0 + h_1.x_2$$

Moreover, let curb weight [lbs] be a function of fuel capacity, engine size, width (WT), height and overall vehicle length:

$$cw = w_0 + w_1.x_1 + w_2.x_2 + w_3.x_3 + w_4.x_6 + w_5.x_7$$

Finally, towing capacity is modeled as:

$$TC = t_0 + t_1.hp + t_2.cw + t_3.x_3 + t_4.x_4 + t_5.x_5 + t_6.x_6$$

Fuel Economy: It is apparent from Fig. B1 (4th column) that vehicles with larger engines have inferior fuel economy. Those vehicles also tend to be heavier. Furthermore, frontal area (width times height) contributes to drag increases even though this can be mitigated by efficient aerodynamic styling. Fuel Economy is therefore approximated as:

$$FE = f_0 + f_1.x_2 + f_2.cw + f_3.x_3.x_6 + +f_4.x_9$$

Acceleration: Assuming constant acceleration we can write

$$v = \ddot{x}.t + v_0 \ and \ \ddot{x} = \frac{F}{m}$$

$$with \ \ P = F.v \ and \ F = \frac{P}{v}$$

the acceleration time from 0-100 km/h is approximated as following:

$$AC = \frac{v^2 m}{P} = (\frac{100000}{3600})^2.cw.\frac{0.45}{746hp}$$

Table D.1: PV parameters across the family

PV	SML	MED	LRG	SPT	SUV	VAN	TRK
Constant	6.07E+03	-1.31E+03	3.80E+03	-6.93E+01	2.72E+03	-1.41E+03	4.63E+03
WT	-5.04E+01	9.90E+00	-2.83E+01	-1.79E+00	-2.02E+01	1.02E+01	-3.60E+01
WB	-3.03E+01	5.92E+00	-1.58E+01	-1.22E-01	-1.51E+01	4.95E+00	-1.49E+01
HT	-5.15E+01	1.23E+01	-3.24E+01	2.68E+00	-1.94E+01	1.13E+01	-3.78E+01
LT	-3.00E-02	4.11E-01	-6.69E-02	3.66E-01	7.76E-01	1.55E+00	-2.07E+00
WT*WB*HT	8.83E-03	-1.64E-03	4.26E-03	1.71E-04	3.16E-03	-1.44E-03	4.85E-03

Table D.2: CV parameters across the family

CV	SML	MED	LRG	SPT	SUV	VAN	TRK
Constant	-4.07E+02	-1.31E+03	-2.20E+03	-8.72E+01	1.24E+03	-7.18E+03	2.32E+03
WT	3.97E+00	9.90E+00	1.72E+01	3.31E-01	-1.12E+01	5.38E+01	-1.93E+01
WB	1.81E+00	5.92E+00	9.14E+00	1.96E-01	-8.03E+00	2.80E+01	-8.83E+00
HT	3.70E+00	1.23E+01	1.89E+01	1.31E+00	-8.77E+00	4.99E+01	-1.78E+01
LT	-1.87E-03	4.11E-01	2.52E-01	8.22E-02	1.53E+00	2.55E+00	2.32E-01
WT*WB*HT	-6.06E-04	-1.64E-03	-2.53E-03	-7.80E-05	1.48E-03	-6.48E-03	2.24E-03

Table D.3: TC parameters across the family

TC	SML	MED	LRG	SPT	SUV	VAN	TRK
Constant	1.09E+04	1.95E+02	6.05E+03	1.19E+04	-1.20E+04	1.28E+03	2.59E+04
FC	-4.83E+01	-1.82E+00	1.74E+02	2.50E+02	2.17E+02	1.56E+03	1.60E+03
ED	1.37E+00	-1.46E-03	6.03E-01	1.71E-01	1.11E+00	-1.47E+00	3.68E+00
WT	-1.36E+01	-1.09E+00	4.08E+02	-6.83E+01	2.13E+02	-5.62E+02	-4.42E+03
WB	-1.15E+02	-2.55E+00	-1.54E+00	3.98E+01	1.91E+01	-1.55E+02	-2.07E+03
GC	-4.10E+01	-3.87E-01	1.04E+02	2.40E+02	2.71E+02	3.15E+03	1.93E+04
HT	-1.19E+02	-1.03E+00	-2.07E+02	1.63E+01	-3.01E+01	-1.70E+02	-1.62E+03
LT	4.11E+01	4.83E-02	-1.20E+02	-9.44E+01	-4.18E+01	1.09E+02	2.01E+03

Table D.4: FE parameters across the family

FE	SML	MED	LRG	SPT	SUV	VAN	TRK
Constant	-2.35E+02	1.95E+02	4.80E+01	-3.85E+02	-1.08E+02	3.69E+01	2.87E+01
FC	-7.88E-01	-1.82E+00	9.48E-02	-3.68E-01	-9.44E-01	-3.19E-01	-3.66E-03
ED	-6.89E-03	-1.46E-03	-7.82E-04	-1.74E-03	-1.27E-03	-1.05E-03	-2.41E-03
WT	5.46E+00	-1.09E+00	-3.74E-01	7.23E+00	1.76E+00	-3.22E-02	4.83E-01
HT	4.50E+00	-2.55E+00	0	7.51E+00	1.80E+00	0	0
LT	6.66E-02	-3.87E-01	-3.91E-02	3.76E-02	2.36E-01	9.72E-04	-4.20E-02
TD	-3.47E-01	-1.03E+00	-1.10E+00	-1.26E+00	-5.01E-01	-2.38E-01	-3.34E-01
WT*HT	-8.82E-02	4.83E-02	6.73E-03	-1.25E-01	-2.74E-02	-2.34E-04	-4.10E-03

Table D.5: HP parameters across the family

HP	SML	MED	LRG	SPT	SUV	VAN	TRK
Constant	-2.75E+01	-5.51E+00	8.17E+01	4.54E+00	5.06E+01	1.64E+01	1.34E+01
	9.05E-02	7.80E-02	4.58E-02	8.85E-02	5.65E-02	5.85E-02	5.70E-02

Table D.6: CW parameters across the family

CW	SML	MED	LRG	SPT	SUV	VAN	TRK
Constant	1.05E+03	5.16E+03	-1.49E+04	-2.81E+03	-4.36E+03	-1.08E+03	7.27E+02
FC	9.60E+01	1.29E+01	6.93E+01	4.31E+01	8.63E+01	-1.83E+01	3.91E+02
ED	2.60E-01	3.53E-01	1.48E-01	2.51E-01	1.49E-01	4.17E-01	5.53E-01
WT	5.80E+01	-3.63E+01	1.18E+02	2.17E+01	1.14E+02	4.84E+01	-1.74E+01
HT	-4.08E+01	-3.60E+01	1.42E+02	5.16E+01	-1.28E+01	-2.96E+00	-4.09E+01
Length	-6.15E+00	7.59E+00	6.05E+00	3.18E+00	-2.66E-01	6.58E+00	-1.45E+01

Appendix E: Automotive Vehicles Raw Data

[1] FE (Fuel Economy is usually indicated by City/Highway. Here it is the averaged value.

[2] WT (Wheel Track) usually has two parameters: front and rear track. But their difference is not very large. Thus here the averaged values of front and rear track are used to indicate the wheel track.

[3] NR means "Not Recommended"

[4] NA means "Not Available"

Table E.1: Small-size sedan (SML)

Brand	Model	PV	CV	TC	FE[1]	AC	FC	ED	WT[2]	WB	GC	HT	LT	TW	TD	HP	CW
Honda	Insight	85.0	15.9	NR[3]	41.5	12.9	10.6	1300	58.4	100.4	NA[4]	56.2	172.3	175	15	98	2723
Honda	Fit	90.8	20.6	NR	31.5	10.2	10.6	1500	58.4	98.4	NA	60.0	161.6	175	15	117	2575
Toyota	Yaris	87.1	12.9	NR	32.0	10.2	11.1	1500	58.1	100.4	5.8	56.7	169.3	175	14	106	2326
Chevrolet	Aveo	90.8	15.0	NR	30.5	11.1	12.0	1600	56.7	97.6	NA	59.3	154.3	185	14	108	2568
Hyundai	Accent	92.2	12.4	NR	30.5	10.2	11.9	1600	57.7	98.4	6.1	57.9	168.5	185	14	110	2403
Kia	Rio	92.2	11.9	NR	30.5	10.2	11.9	1600	57.7	98.4	6.1	57.9	166.9	185	14	110	2403
Kia	Rio5	92.2	15.8	NR	30.5	10.5	11.9	1600	57.7	98.4	6.1	57.9	158.1	185	14	110	2487
Honda	Civic	90.9	12.0	NR	30.5	8.9	13.2	1800	59.6	106.3	6.1	56.5	177.3	195	15	140	2692
Nissan	Versa	94.7	14.0	NR	28.0	10.6	13.2	1800	58.4	102.4	6.3	60.4	169.1	185	15	122	2780
Scion	xD	84.5	10.5	NR	29.0	9.7	11.1	1800	58.6	96.9	6.5	60.0	154.7	195	16	128	2665
Toyota	Corolla	92.0	16.6	1500	30.0	9.7	13.2	1800	60.4	102.4	5.8	57.7	178.7	195	15	132	2745
Audi	A4	72.9	12.0	NR	26.5	7.8	17.2	2000	59.9	110.6	4.2	56.2	185.2	225	17	211	3527
Ford	Focus	93.4	13.8	NR	29.5	8.6	13.5	2000	58.2	102.9	6.2	58.6	175.0	195	15	140	2588
Hyundai	Elantra	97.9	14.2	750	29.0	9.3	14.0	2000	60.7	104.3	5.9	58.3	177.4	195	15	138	2747

Kia	Forte	96.8	14.7	NR	29.5	8.2	13.7	2000	60.9	104.3	5.9	57.5	178.3	195	15	156	2740
Kia	Spectra	97.0	18.3	NR	28.0	9.7	14.0	2000	58.7	102.8	6.3	57.9	171.3	205	16	138	2886
Mitsubishi	Lancer	93.5	12.3	NR	25.0	9.2	15.3	2000	60.2	103.7	5.9	58.7	180.0	205	16	152	2999
Nissan	Sentra	97.4	13.1	NR	29.0	9.7	14.5	2000	60.3	105.7	5.6	59.5	179.8	205	15	140	2930
Suzuki	SX4 sport	88.5	16.0	NR	26.0	8.7	13.2	2000	59.0	98.4	6.3	60.8	177.6	205	17	143	2668
Volkswagen	GLI	91.0	16.0	NR	25.5	7.8	14.5	2000	60.2	101.5	5.4	57.4	179.3	225	17	200	3334
Volkswagen	GTI	94.2	15.1	NR	25.5	7.5	14.5	2000	60.1	101.5	5.6	58.4	165.8	225	17	200	3213
Chevrolet	Cobalt	83.0	13.9	1000	28.5	8.4	13.0	2200	57.6	103.3	5.4	57.1	180.3	195	15	155	2783
Acura	TSX	94.5	12.6	1000	25.5	8.0	18.5	2400	62.2	106.4	5.9	56.7	185.6	225	17	201	3470
Volvo	S40	92.5	12.6	2000	24.0	9.1	15.9	2400	60.4	103.9	5.3	57.2	176.2	205	17	168	3273
Lexus	IS	88.3	13.0	NR	25.0	7.8	17.1	2500	60.4	107.5	5.7	56.1	180.3	225	17	204	3435
Mazda	Mazda3	94.6	17.0	NR	25.5	8.5	15.9	2500	59.9	103.9	4.7	57.9	177.4	205	17	167	3064
Subaru	Impreza	94.4	11.3	2000	23.0	8.7	16.9	2500	58.9	103.1	6.1	58.1	180.3	205	16	170	3163
Subaru	Impreza WRX	94.4	11.3	998	21.5	5.6	16.9	2500	59.0	103.1	6.1	58.1	180.3	225	17	265	3174
Volkswagen	Jetta	91.0	16.0	2000	24.5	9.0	14.5	2500	60.2	101.5	5.4	57.4	179.3	205	16	170	3285
Volkswagen	Rabbit	94.2	15.1	NR	24.5	8.6	14.5	2500	60.2	101.5	5.4	58.2	165.8	195	15	170	3138
Saab	9 3	93.4	15.0	2000	19.5	6.5	16.0	2800	59.7	105.3	5.9	57.1	182.9	235	17	255	3570

Table E.2: Mid-sized (MED) sedan

Brand	Model	PV	CV	TC	FE[1]	AC	FC	ED	WT[2]	WB	GC	HT	LT	TW	TD	HP	CW
Nissan	Altima	100.7	15.3	1000	27.0	8.5	20.0	2500	60.85	109.3	5.2	57.9	189.8	215	16	175	3179
Honda	Accord	106.0	14.0	1000	25.5	8.6	18.5	2400	62.20	110.2	5.7	58.1	194.1	215	16	177	3289
Pontiac	G6	110.1	14.0	1000	26.0	9.4	16.0	2400	60.10	112.3	6.5	57.1	189.1	215	17	164	3305
Toyota	Camry	101.4	15.0	1000	27.0	9.1	18.5	2500	61.80	109.3	5.5	57.9	189.2	215	16	169	3307
Chrysler	Sebring	102.5	13.6	1000	25.5	8.9	16.9	2400	61.80	108.9	5.3	59.0	190.6	215	16	173	3310
Ford	Fusion	100.3	16.5	1000	26.5	8.9	17.5	2500	61.50	107.4	5.5	56.9	190.6	225	17	175	3342
Saturn	Aura	98.0	14.9	1000	27.5	9.5	16.0	2400	60.80	112.3	6.2	57.6	190.9	215	17	169	3444
Chevrolet	Impala	104.5	18.6	1000	24.0	7.8	17.0	3500	61.95	110.5	5.9	58.7	200.4	225	16	211	3555
Nissan	Maxima	96.2	14.2	1000	22.5	5.7	20.0	3500	62.40	109.3	5.5	57.8	190.6	245	18	290	3556
Acura	TL	98.2	13.1	1000	22.0	6.2	18.5	3500	63.50	109.3	5.9	57.2	195.3	245	17	280	3708
Cadillac	CTS	98.0	13.6	1000	21.5	5.9	18.0	3600	62.05	113.4	4.5	58.0	191.6	235	17	304	3874
Buick	LaCrosse	99.4	13.3	1000	22.5	7.2	18.0	3000	61.85	111.7	5.9	59.2	196.9	225	16	255	3948
Acura	RL	99.1	13.1	1000	19.0	6.3	19.4	3700	62.20	110.2	5.7	57.2	195.8	245	18	300	4083
Chevrolet	Malibu	95.0	15.1	1000	26.0	9.4	16.0	2400	59.80	112.3	5.5	57.1	191.8	215	17	169	3415
Dodge	Avenger	100.9	13.4	1000	25.5	9.0	16.9	2400	61.80	108.9	5.2	58.9	190.9	215	16	173	3332
Mercury	Sable	108.0	21.2	1000	23.0	6.5	20.5	3500	64.35	112.9	5.1	61.5	202.1	215	17	260	3643
Hyundai	Sonata	105.4	16.3	1500	27.0	8.8	17.7	2400	61.70	107.4	6.3	58.0	188.9	215	16	175	3327
Lincoln	MKZ	99.0	16.5	1500	22.5	6.4	17.5	3500	61.50	107.4	NA	56.9	189.8	225	17	263	3598
Jaguar	XF series	99.3	17.7	1653	20.0	6.2	18.4	4200	62.30	114.5	4.1	57.5	195.3	245	18	300	4017
Kia	Optima	104.2	15.0	2000	27.0	8.5	16.4	2400	61.30	107.1	6.3	58.3	189.0	205	16	175	3197
Audi	A6	97.9	15.9	2000	25.0	7.0	21.1	3200	63.60	111.9	4.6	57.4	193.5	245	17	255	3858
Subaru	Legacy	103.0	14.7	2700	23.0	9.3	18.5	2500	61.60	108.3	5.9	59.3	186.4	205	16	170	3388
Mitsubishi	Galant	100.2	13.3	2866	23.5	10.1	17.7	2400	61.80	108.3	6.0	57.9	191.0	215	16	160	3483
Volvo	S60	94.0	13.9	3300	23.5	7.9	18.0	2500	61.10	106.9	5.2	56.2	181.2	235	17	208	3523
Lexus	GS 450h	110.5	10.6	3500	23.5	6.6	17.2	3500	60.50	112.2	5.1	56.1	190.0	245	18	292	4134

Saab	9 5	96.2	15.9	3500	22.0	6.2	18.0	2300	59.90	106.4	6.6	57.2	190.4	235	17	260	3470
Volkswagen	Passat	96.3	14.2	4000	24.0	7.8	18.5	2000	61.10	106.7	5.2	58.0	188.2	235	17	200	3344
Volkswagen	CC	96.3	13.0	4000	24.0	7.9	18.5	2000	61.25	106.7	5.0	55.8	188.9	235	17	200	3374
Mercedes-Benz	C-Class	88.2	12.4	4000	21.5	7.3	17.4	3000	60.00	108.7	4.5	56.9	182.3	225	17	228	3560
Toyota	Camry	101.4	15.0	1000	27.0	9.1	18.5	2500	61.80	109.3	5.5	57.9	189.2	215	16	169	3307
Mercury	Milan	100.3	16.5	NA	26.5	8.8	17.5	2500	61.45	107.4	6.8	56.9	189.0	225	17	175	3308
Mazda	Mazda6	101.9	16.6	NA	25.5	9.1	18.5	2500	62.80	109.8	5.1	57.9	193.7	205	16	170	3309
Lexus	ES 350	95.4	14.7	NA	23.0	6.1	18.5	3500	61.95	109.3	6.1	57.1	191.1	215	17	272	3580
Toyota	Prius	93.7	21.6	NR	49.5	14.4	11.9	1800	59.90	106.3	5.5	58.7	175.6	195	15	98	3042

Table E.3: Large-size sedan (LRG)

Brand	Model	PV	CV	TC	FE[1]	AC	FC	ED	WT[2]	WB	GC	HT	LT	TW	TD	HP	CW
Chrysler	300	107.0	15.6	1000	22.0	9.7	18.0	2700	63.05	120.0	5.6	58.4	196.8	215	17	178	3725
Dodge	Charger	104.0	16.2	1000	22.0	9.1	18.0	2700	63.05	120.0	5.2	58.2	200.1	215	17	190	3728
Hyundai	Azera	106.9	16.6	2000	22.0	7.2	19.8	3300	61.90	109.4	6.4	58.7	192.7	235	17	234	3629
Ford	Taurus	102.3	20.1	1000	23.0	7.7	19.0	3500	64.35	112.9	5.1	60.7	202.9	235	17	263	4368
Toyota	Avalon	106.9	14.4	1000	23.5	6.1	18.5	3500	61.9	111	5.3	58.5	197.6	215	16	268	3505
Pontiac	G8	107.0	17.5	2000	21.0	7.1	19.0	3600	63.00	114.8	5.5	57.7	196.1	245	18	256	3885
Buick	Lucerne	108.0	17.0	1000	21.5	7.7	18.0	3880	62.75	115.6	5.1	58.0	203.2	235	17	227	3735
Audi	A8	106.8	14.6	3417	19.5	5.7	23.8	4200	63.85	115.9	4.7	57.3	199.3	255	18	350	4321
Jaguar	XJ Series	104.6	16.4	1654	20.5	5.8	22.5	4200	61.10	119.4	5.1	57.0	200.4	235	18	300	3770
Cadillac	DTS	105.5	18.8	1000	19.0	6.8	18.0	4600	62.10	115.6	5.3	57.6	207.6	235	17	275	4009
Mercury	Grand Marquis	107.5	20.6	1500	20.0	7.9	19.0	4600	64.50	114.6	5.2	56.3	212.0	225	17	224	3796

Table E.4: Sports car (SPT)

Brand	Model	PV	CV	TC	FE[1]	AC	FC	ED	WT[2]	WB	GC	HT	LT	TW	TD	HP	CW
Audi	RS 4	90.0	13.4	1700	16.5	4.4	16.6	4163	58.90	104.3	3.7	55.7	180.7	255	19	420	3957
Audi	S4	73	10.2	NR	16	5.8	16.6	4163	59.9	104.5	4	54.8	180	235	18	340	4211
Audi	S5	84.0	16.1	1900	18.5	5.2	16.6	4163	62.35	108.2	4.1	53.9	182.5	255	19	354	3935
Audi	S6	97.7	15.9	NR	16.5	4.8	21.1	5204	62.40	112.1	4.2	57.0	193.5	265	19	435	4486
Honda	S2000	45.1	5	1000	21.5	5.6	13.2	2200	58.65	94.5	4.2	50	162.7	215	17	237	2864
Porsche	Cayman	48.0	14.4	NR	22.0	4.3	16.9	3400	59.50	95.1	4.3	51.4	171.1	235	18	320	2976
Mazda	MX-5 Miata	46.0	5.3	NR	25.0	6.9	12.7	2000	58.80	91.7	4.6	49.0	157.3	205	17	167	2480
Mazda	RX-8	89.0	7.6	NR	19.0	6.1	16.9	1308	59.20	106.4	4.7	52.8	175.6	225	18	232	3064
Mercedes-Benz	SLK	49.0	9.8	NR	22.0	5.1	18.5	3498	60.45	95.7	4.7	51.1	161.5	225	17	300	3318
Lotus	Elise	41.0	4.1	NR	24.0	4.9	10.6	1797	58.30	90.6	5.0	45.0	149.0	225	17	189	1984
Dodge	Challenger	94.0	16.2	1000	21.0	6.9	18.5	3500	63.05	116.0	5.1	57.3	197.7	215	16	250	3719
Scion	tC	84.7	12.8	NR	23.5	8.4	14.5	2382	61.75	106.3	5.2	55.7	174.0	215	17	161	2905
Aston Martin	V8 Vantage	70.0	10.6	3500	15.0	4.0	21.1	4735	61.75	102.5	5.3	49.5	172.5	275	19	420	3595
Chevrolet	Cobalt	86.0	13.9	1000	31.0	8.2	13.0	2200	57.60	103.3	5.3	55.5	180.5	205	16	155	2721
Volvo	C30	89.0	15.3	2000	25.0	6.6	15.9	2500	60.35	103.9	5.3	57.0	167.4	205	17	227	3201
BMW	Z4	57.0	10.9	3527	23.5	5.9	14.5	3000	58.15	98.3	5.4	50.8	166.9	225	18	255	3241
Hyundai	Tiburon	81.9	14.8	1000	24.0	9.8	14.5	2000	58.70	99.6	5.4	52.4	173.0	205	16	138	2898
Mitsubishi	Eclipse	82.0	14.8	NR	24.0	9.4	17.7	2400	61.80	101.4	5.8	53.5	180.4	225	17	162	3269
Volkswagen	GLI	91.0	16.0	NR	25.0	7.7	14.5	1984	60.20	101.5	5.4	57.4	179.3	225	18	200	3290
Ford	Shelby GT500	96.0	13.4	1000	18.0	3.4	16.0	5400	62.20	107.1	5.7	55.6	188.1	225	18	540	3920
Lexus	IS F	85.7	13.3	NR	19.5	4.2	16.9	4968	60.50	107.5	5.7	55.7	183.5	225	19	416	3780
Audi	TT	74.0	13.1	3307	27.0	6.9	14.5	1968	61.60	97.2	4.4	53.2	164.5	245	18	200	2965
BMW	M3	93.0	11.1	NR	17.0	4.2	16.6	3999	60.60	108.7	6.5	55.8	181.8	245	19	414	3704
BMW	M5	99.0	14.0	1654	14.0	3.7	18.5	4999	61.95	113.7	4.5	57.8	191.5	255	19	500	4012
BMW	M6	82.0	13.0	NR	14.0	3.6	18.5	4999	59.10	109.5	4.1	54.0	191.8	255	19	500	3909

BMW	Z4 M		47.6	10.6	NR	18.5	4.5	14.5	3246	59.10	98.3	4.0	51.3	161.9	225	18	330	3197
Chevrolet	Camaro		81.9	11.3	1000	23.0	5.8	19.0	3600	63.90	112.3	4.0	54.2	190.4	245	19	304	3780
Lotus	Exige		41.0	4.0	NR	23.0	4.0	10.6	1796	58.35	90.5	5.1	45.6	149.5	195	16	240	2077
Nissan	GT-R		79.0	8.8	NR	18.5	3.7	19.5	3799	62.80	109.4	4.3	54.0	183.1	285	20	485	3814
Nissan	350Z		51.8	6.8	NR	21.5	5.1	20.0	3498	60.55	104.3	4.7	52.1	169.8	225	18	306	3339
Pontiac	Solstice		49.0	5.4	NR	22.0	7.7	13.0	2384	61.05	95.1	5.0	50.1	157.2	245	18	173	2860
Porsche		911	70.0	3.7	NR	18.5	3.9	17.7	3797	59.10	92.5	4.3	51.2	175.6	235	18	385	3252
Porsche	Boxster		NA	9.9	NR	23.0	5.4	16.9	2900	59.55	95.1	4.1	50.9	172.1	235	18	255	2943
Saturn	Sky		50.0	5.4	1000	22.0	8.0	13.0	2384	61.10	95.1	3.6	50.2	161.1	245	18	173	2965
Subaru	Impreza		94.4	19.0	2700	23.5	8.4	16.9	2457	58.40	103.1	6.5	58.1	173.8	205	16	170	3064
Chrysler	Crossfire		49.0	7.6	NR	19.0	6.6	15.9	3199	59.10	94.5	4.9	51.5	159.8	225	18	215	3061
Ford	Mustang		81.6	13.4	1000	22.0	7.5	16.0	4099	62.90	107.1	5.7	55.6	188.1	235	18	210	3401
Mazda	MAZDASPEED3		95.0	17.0	NR	21.5	5.7	15.9	2260	60.20	103.9	4.8	57.5	177.6	205	17	263	3245
Volkswagen	R32		93.1	9.7	NR	20.5	6.6	14.5	3189	60.00	101.5	5.6	57.7	167.2	225	18	250	3547

Table E.5: Sport Utility Vehicle (SUV)

Brand	Model	PV	CV	TC	FE[1]	AC	FC	ED	WT[2]	WB	GC	HT	LT	TW	TD	HP	CW
Volkswagen	Tiguan	95.3	23.6	2200	21.0	8.0	16.8	2000	61.80	102.5	6.9	66.3	174.3	215	16	200	3433
Kia	Sportage	103.9	23.6	1500	22.5	10.8	17.2	2200	60.60	103.5	7.7	66.7	171.3	215	16	140	3254
Acura	RDX	101.4	27.8	1500	19.5	7.6	18.0	2300	62.25	104.3	6.3	65.1	180.7	235	18	240	3931
Mazda	CX-7	101.7	29.9	2000	20.0	7.1	18.2	2300	63.70	108.3	8.1	64.8	184.0	235	18	244	3710
Chevrolet	Equinox	99.7	32.0	1500	27.0	9.6	18.0	2400	62.15	112.5	7.8	66.3	187.8	225	17	182	3761
Honda	CR-V	100.9	30.9	1500	23.5	9.5	15.3	2400	61.60	103.1	7.3	66.1	177.9	225	17	166	3389
Honda	Element	103.6	25.1	1500	22.5	9.9	15.9	2400	62.20	101.4	6.9	70.7	169.9	215	16	166	3515
Jeep	Compass	101.3	21.9	1000	25.5	8.7	13.5	2400	59.80	103.7	8.1	65.2	173.4	215	17	172	3223
Jeep	Patriot	101.7	23.0	2000	25.5	8.8	13.5	2400	59.80	103.7	8.0	65.7	173.6	205	16	172	3250
Mitsubishi	Outlander	100.4	39.0	1500	22.5	9.4	16.6	2400	59.60	105.1	8.5	66.1	182.7	215	16	168	3395
Saturn	Vue	100.0	30.8	1500	20.5	10.1	19.0	2400	61.80	106.6	7.8	67.0	180.1	235	16	169	3664
Suzuki	Grand Vitara	106.7	24.4	3000	22.0	10.9	17.4	2400	61.00	103.9	7.4	66.7	177.1	225	16	166	3876
Ford	Escape	99.5	29.2	1500	22.0	9.1	15.1	2500	60.45	103.1	8.4	67.8	174.7	235	16	171	3355
Mazda	Tribute	99.4	29.3	1500	24.0	9.1	16.5	2500	60.60	103.1	8.3	67.9	174.9	235	16	171	3357
Mercury	Mariner	99.4	29.3	3500	24.0	9.2	15.0	2500	60.45	103.1	8.4	68.6	175.2	235	16	171	3385
Nissan	Rogue	98.0	28.9	1000	24.5	9.0	15.9	2500	60.80	105.9	8.3	65.3	182.9	215	16	170	3281
Subaru	Forester	107.6	33.5	2400	22.0	8.9	15.9	2500	60.20	103.0	8.7	65.9	179.5	215	16	170	3250
Toyota	RAV4	108.2	36.4	1500	25	8.8	15.9	2500	61	104.7	7.5	66.3	181.9	215	16	179	3360
Hyundai	Tucson	102.6	22.7	2000	21.0	9.1	17.2	2700	61.00	103.5	7.7	68.1	170.3	235	16	173	3370
BMW	X3	90.1	30.0	3500	19.5	7.2	17.7	3000	60.35	110.1	8.0	65.9	179.9	235	17	260	4012
BMW	X5	102.4	21.9	6000	18.0	8.8	22.5	3000	64.85	115.5	8.3	69.9	191.1	255	18	260	4915
BMW	X6	98.0	25.6	6000	17.5	7.6	22.5	3000	65.95	115.5	8.5	66.5	192.0	315	20	300	4894
Cadillac	SRX	100.6	32.4	3500	21.5	7.4	21.0	3000	63.85	110.0	7.0	65.7	190.3	235	18	265	4204
Audi	Q5	101.5	27.3	4400	20.5	7.3	19.8	3200	63.60	110.5	7.9	65.1	182.2	235	18	270	4244
Land Rover	LR2	102.4	26.7	3500	18.5	8.6	17.7	3200	63.50	104.7	8.3	68.5	177.1	235	19	230	4255

Kia	Sorento	105.8	31.7	3500	19.0	7.8	21.1	3300	62.20	106.7	8.2	68.1	180.7	245	16	242	4068
Pontiac	Torrent	102.6	32.5	3500	20.5	9.2	20.0	3400	61.60	112.5	8.0	69.3	188.8	235	16	185	3660
Buick	Enclave	154.0	65.9	4500	20.5	7.7	22.0	3600	67.20	118.9	8.4	72.2	201.8	255	19	288	4780
Jeep	Liberty	104.1	31.5	5000	19.0	8.9	19.5	3700	61.00	106.1	7.4	70.6	176.9	225	16	210	4030
Acura	MDX	142.2	42.9	5000	17.5	7.1	21.0	3700	67.60	108.3	8.2	68.2	190.7	255	18	300	4548
Jeep	Wrangler	102.9	17.2	3500	17.5	9.2	21.6	3800	61.90	116.0	8.7	70.8	183.0	225	16	202	3976

Table E.6: Van/Minivan

Brand	Model	PV	CV	TC	FE[1]	AC	FC	ED	WT[2]	WB	GC	HT	LT	TW	TD	HP	CW
Ford	Transit Connect	167.8	135.3	1600	23.5	11.5	15.1	1999	60.20	114.6	7.9	79.3	180.7	205	15	136	3360
Dodge	Grand caravan	182.0	143.8	1800	20.5	10.9	20.0	3300	65.15	121.2	6.1	68.9	202.5	225	16	175	4091
Toyota	Sienna	177.4	140.6	3500	20	7.5	20	3500	66.25	119.3	6.9	68.9	201	215	16	265	4270
Chrysler	Town & Country	163.5	143.8	1800	20.5	11.5	20.0	3301	65.90	121.2	6.3	68.9	202.5	225	16	175	4335
Chevrolet	Uplander	170.1	120.1	3500	19.5	8.2	20.0	3900	62.40	113.0	6.5	70.5	191.0	225	17	240	4233
Volkswagen	Routan	163.5	140.6	3500	19.5	10.6	20.5	3800	65.00	121.2	6.4	68.9	202.5	225	16	197	4507
Honda	Odyssey	171.4	147.4	3500	19.5	8.4	21.0	3471	66.70	118.1	6.3	68.8	202.1	235	16	244	4387
Nissan	Quest	171.4	147.4	3500	19.5	8.4	21.0	3498	67.30	118.1	6.3	68.8	202.1	235	16	244	4387
Hyundai	Entourage	172.3	141.5	3500	19.5	8.2	21.1	3778	66.30	118.9	6.6	71.5	202.0	225	16	250	4400
Kia	Sedona	160.5	121.3	3500	19.5	8.3	21.1	3778	66.30	113.8	6.6	69.3	189.4	225	16	244	4365

Table E.7: Trunk (TRK)

Brand	Model	PV	CV	TC	FE[1]	AC	FC	ED	WT[2]	WB	GC	HT	LT	TW	TD	HP	CW
Chevrolet	Colorado	56.0	44.0	4000	21.0	8.4	19.0	2900	57.5	111.3	7.3	64.9	192.4	225	15	185	3337
Dodge	Dakota	94.3	45.8	7100	17.5	9.5	22.0	3700	62.8	131.3	7.9	68.0	218.8	245	16	210	4296
Ford	Explorer Sport Trac	106.1	44.4	5090	16.0	8.7	22.5	4000	60.8	130.5	8.5	72.5	210.2	235	16	254	4740
Ford	Ranger	52.5	37.4	6000	21.5	9.9	20.0	2300	58.5	118.0	7.6	66.2	200.5	225	15	143	3028
GMC	Canyon	55.0	42.6	4000	21.0	8.4	19.0	2900	59.7	111.3	7.9	65.5	192.4	225	15	185	3337
Isuzu	I290 s extended cab short bed	87.7	44.0	2900	21.0	8.8	19.5	2900	57.5	126.0	7.3	64.9	207.1	225	15	185	3488
Mazda	B-series	63.7	37.0	5600	17.0	8.2	19.5	4000	58.0	125.9	7.9	67.5	202.9	235	15	207	3662
Mitsubishi	Raider	102.3	42.4	4000	17.5	9.6	22.0	3700	62.9	131.3	7.9	68.6	218.5	265	16	210	4311
Nissan	Frontier	87.7	33.5	6300	17.5	7.6	21.1	4000	61.8	125.9	8.6	70.1	205.5	265	16	261	4274
Suzuki	Equator	101.1	33.5	6300	17.5	7.6	21.1	4000	61.8	125.9	8.6	70.1	206.6	265	16	261	4248
Toyota	Tacoma	89	39.7	3500	22	10.5	21	2700	61	127.2	8.1	65.7	208.1	215	15	159	3590

Table E.8: Production volume

Model	Annual Demand
SML	32,241
MED	91,464
LRG	35,556
SPT	40,074
SUV	27,681
VAN	209,727
TRK	363,018

Appendix F: Utility Functions

F.1: Utility functions for Objectives:

Table F.1: Objective attributes

Objective	EA	Model	Type
f1	FE	SML	Largest is Best
f2	PV	MED	Value is Best
f3	CV	VAN	Largest is Best

FE in SML: Largest-is-Best type of utility function. $f_{min}=19$ and f_{max} is 41.5.

$$u(FE_{SML}) = \begin{cases} 0, & f \leq 19.5 \\ \dfrac{f - 19.5}{41.5 - 19.5}, & 19.5 \leq f \leq 41.5 \\ 0, & f > 41.5 \end{cases}$$

PV in MED: Value-is-Best type of utility function. The desired value is assumed to be the average value for products in the same market segment $f_{D-val} = 100.4$ cft. $f_{min}=88.20$ cft and $f_{max}=110.50$ cft

$$u(PV_{MED}) = \begin{cases} 0, & f \leq 88.2 \\ \dfrac{f - 88.2}{100.4 - 88.2}, & 88.2 \leq f \leq 100.4 \\ \dfrac{110.5 - f}{110.5 - 100.4}, & 100.4 \leq f \leq 110.5 \\ 0, & f > 110.5 \end{cases}$$

CV in VAN: Largest-is-Best type of utility function. $f_{min}=120.1$ cft and $f_{max}=147.4$ cft

$$u(CV_{VAN}) = \begin{cases} 0, & f \leq 120.1 \\ \dfrac{f - 120.1}{147.4 - 120.1}, & 120.1 \leq f \leq 147.4 \\ 0, & f > 147.4 \end{cases}$$

F.2: Utility functions for constraints:

PV in SML: Acceptable level: 0.7

$$u(PV_{SML}) = \begin{cases} 0, & f \le 72.90 \\ \dfrac{f - 72.90}{91.46 - 72.90}, & 72.90 \le f \le 91.46 \\ \dfrac{97.9 - f}{97.9 - 91.46}, & 91.46 \le f \le 97.9 \\ 0, & f > 110.5 \end{cases}$$

CV in MED: Range-is-Best, f_{min}=10.6, f_{max}=21.60 cft, f_l=18 F_u=20, Acceptable level 0.8

$$(f) = \begin{cases} 0, & f \le 10.6 \\ \dfrac{f - 10.6}{18 - 10.6}, & 10.6 \le f \le 18 \\ 1, & 18 \le f \le 20 \\ \dfrac{21.6 - f}{21.6 - 20}, & 20 \le f \le 21.6 \\ 0, & f > 21.6 \end{cases}$$

Acceleration in SPT: Value-is-Best, f_{min}=3.38, f_{max}=9.77, $f_{D\text{-}value}$=5.87, Acceptable level 0.7

$$u(f) = \begin{cases} 0, & f \le 3.38 \\ \dfrac{f - 3.38}{5.87 - 3.38}, & 3.38 \le f \le 5.87 \\ \dfrac{9.77 - f}{9.77 - 5.87}, & 5.87 \le f \le 9.77 \\ 0, & f > 9.77 \end{cases}$$

Appendix G: CAF^{2}M results

The Analytic Hierarchy Process (AHP) enables decision makers to structure decisions hierarchically with the overall goal of the decision at the top of the model, strategic objectives in the higher levels, evaluation criteria in the middle levels, and alternative choices at the bottom.

The AHP provides a structured framework for setting priorities on each level of the hierarchy using pairwise comparisons, a process of evaluating each pair of decision factors at a given level on the model for their relative importance with respect to their parent.

The consistency of the judgments is tracked using the rigorous math analytics behind the AHP to validate the decision process. In cases where inconsistency ratio is above 10% it is recommended that the criteria and judgments be revisited. More information about AHP can be found in Satty (1980).

Table G.1: Pairwise-comparison matrix sample (n=8, RI=1.41)

Alternative	1	2	3	4	5	6	7	8
1	1							
2	3	1						
3	4	2	1					
4	3	5	1	1				
5	9	3	6	7	1			
6	8	7	6	5	3	1		
7	7	4	5	7	4	3	1	
8	2	3	5	6	7	9	6	1

Table G.2: Pairwise-comparison matrix for Decision Maker 1 (N=8, RI=1.41)

Alternative	1	2	3	4	5	6	7	8
1	1	1/3	1/4	1/6	1/5	1/7	1/8	1/9
2	3	1	1/2	1/3	1/2	1/4	1/3	1/6
3	4	2	1	1/3	1	1/2	1/3	1/7
4	6	3	3	1	2	1	1/2	1/4
5	5	2	1	1/2	1	1/3	1/4	1/6
6	7	4	2	1	3	1	1	1/4
7	8	3	3	2	4	1	1	1/3
8	9	6	7	4	6	4	3	1

Maximal eigen-value=8.3038, CI= 0.0434, CR= 0.03078<0.1

Table G.3: Pairwise-comparison matrix for Decision Maker 2 (N=8, RI=1.41)

DM2								
Alternative	1	2	3	4	5	6	7	8
1	1	1/4	1/5	1/5	1/6	1/8	1/7	1/9
2	4	1	1/3	1/3	1/3	1/5	1/4	1/7
3	5	3	1	1/4	1/2	1/3	1/3	1/6
4	5	3	4	1	2	1/2	1/3	1/5
5	6	3	2	1/2	1	1/3	1/4	1/6
6	8	5	3	2	3	1	1/2	1/4
7	7	4	3	3	4	2	1	1/4
8	9	7	6	5	6	4	4	1

Maximal eigen-value=8.6817, CI= 0.097386, CR= 0.069068<0.1

Table G.4: Pairwise-comparison matrix for Decision Maker 3 (N=8, RI=1.41)

Alternative	1	2	3	4	5	6	7	8
1	1	1/3	1/4	1/6	1/5	1/7	1/8	1/9
2	3	1	1/3	1/4	1/4	1/4	1/5	1/7
3	4	3	1	1/3	1/3	1/4	1/4	1/5
4	6	4	3	1	3	1	1/2	1/4
5	5	4	3	1/3	1	1/3	1/4	1/6
6	7	4	4	1	3	1	1/2	1/4
7	8	5	4	2	4	2	1	1/4
8	9	7	5	4	6	4	4	1

Maximal eigen-value=8.6257, CI= 0.089386, CR= 0.069068<0.1

Table G.5: Pairwise-comparison matrix for Decision Maker 4 (N=8, RI=1.41)

Alternative	1	2	3	4	5	6	7	8
1	1	1/3	1/4	1/6	1/6	1/6	1/8	1/9
2	3	1	1/3	1/4	1/4	1/5	1/5	1/7
3	4	3	1	1/3	1/3	1/4	1/4	1/6
4	6	4	3	1	3	1	1/2	1/4
5	6	4	3	1/3	1	1/3	1/4	1/6
6	6	5	4	1	3	1	1/2	1/4
7	8	5	4	2	4	2	1	1/4
8	9	7	6	4	6	4	4	1

Maximal eigenvalue=8.6343, CI= 0.090614, CR= 0.069068<0.1

Table G.6: AOF parameters for Decision Makers

	DM 1	DM 2	DM 3	DM 4	Average
FE	0.19	0.17	0.17	0.17	0.17
PV	0.23	0.23	0.24	0.24	0.24
CV	0.25	0.28	0.27	0.27	0.27
FE*CV	0.09	0.09	0.10	0.10	0.10
FE*CV	0.11	0.11	0.09	0.09	0.10
PV*CV	0.12	0.13	0.12	0.12	0.12

Appendix H: Optimization Results

Table H.1: Optimization results for Platform Scenario 1

Market segment	PV	CV	TC	FE	AC	FC	ED	WT	WB	GC	HT	LT	TW	TD	HP	CW
SML	102.5	14.6		36.3	12.9	10.6	1312	62.2	109.4	6.3	58.7	175.1	175	14	91.2	2535
MED	106.3	16	1000	42.2	10.3	13.1	1800	62.2	109.4	6.3	57	175.6	215	16	134.9	2984
LRG	128.3	24	2782	23.2	8.3	23.6	2700	62.2	109.4	6.3	56.4	192.7	215	16	205.4	3641
SPT	71.9	8.2		31	7.2	10.6	1805	62.2	109.4	6.3	48.5	191.3	195	16	164.3	2556
SUV	112.2	28.9	3173	17.5	10.1	22.5	2500	62.2	109.4	6.3	64.8	180.9	215	16	191.9	4183
VAN	182	122	1600	21.4	11.5	20	2437	67.3	121.2	6.1	68.8	199.1	215	16	159	3937
TRK	52	25	2900	21.8	9.2	19	2300	57.5	122	7.6	64.9	201.2	215	16	144.5	2849

Table H.2: Utility Value for Platform Scenario 1

	Model	Utility Value
AOF:		0.485
PV	SML	1
	MED	0.763
CV	MED	1
	VAN	0.256
FE	SML	0.771
AC	SPT	1

Table H.3: Optimization results for Platform Scenario 2

Market segment	PV	CV	TC	FE	AC	FC	ED	WT	WB	GC	HT	LT	TW	TD	HP	CW
SML	102.5	14.3	.	36.5	12.9	10.6	1316	62.2	110.6	6.3	58.5	175.1	175	14	91.6	2546
MED	105.9	16	1000	42.2	10.4	11.9	1802	61.1	109.4	6.4	56.7	175.6	215	16.3	135	3018
LRG	130	22.4	2890	22	6.7	23.8	3836	61.1	109.4	6.4	56.4	194.1	215	16.3	257.4	3705
SPT	45.1	2.5	.	24.4	3.4	10.6	4466	61.1	109.4	6.4	45	154.4	195	16	399.8	2902
SUV	104.9	17.2	4210	17.5	7.1	19	3871	62.2	110.6	6.3	64.8	174.9	215	16.3	269.4	4084
VAN	182	122	1600	21	10.8	20.2	2706	67.3	110.6	6.1	68.8	199.1	215	16.3	174.6	4045
TRK	52	25	2900	21.9	10.5	19.8	2300	57.5	121.2	7.9	65.3	192.4	215	15	144.5	3262

Table H.4: Utility value for Platform Scenario 2

	Model	Utility Value
AOF:		0.485
PV	SML	1
	MED	0.763
CV	MED	1
	VAN	0.256
FE	SML	0.771
AC	SPT	0.991

Table H.5: Optimization results for Platform Scenario 3

Market segment	PV	CV	TC	FE	AC	FC	ED	WT	WB	GC	HT	LT	TW	TD	HP	CW
SML	102.5	14.6		36.3	12.9	10.6	1312	62.2	109.4	6.3	58.7	175.1	175.0	14.0	91.2	2535
MED	106.3	16.0	1000	42.2	10.3	13.1	1800	62.2	109.4	6.3	57.0	175.6	215.0	16.0	134.9	2984
LRG	128.3	24.0	2782	23.2	8.3	23.6	2700	62.2	109.4	6.3	56.4	192.7	215.0	16.0	205.4	3641
SPT	71.9	8.2		31.0	7.2	10.6	1805	62.2	109.4	6.3	48.5	191.3	195.0	16.0	164.3	2556
SUV	112.2	28.9	3173	17.5	10.1	22.5	2500	62.2	109.4	6.3	64.8	180.9	215.0	16.0	191.9	4183
VAN	182.0	122.0	1600	21.4	11.5	20.0	2437	67.3	121.2	6.1	68.8	199.1	215.0	16.0	159.0	3937
TRK	52.0	25.0	2900	21.8	9.2	19.0	2300	57.5	122.0	7.6	64.9	201.2	215.0	16.0	144.5	2849

Table H.6: Utility value for Platform Scenario 3

	Model	Utility Value
AOF:		0.478
PV	SML	1
	MED	0.75
CV	MED	1
	VAN	0.256
FE	SML	0.762
AC	SPT	0.704

Table H.7: Optimization results for Platform Scenario 4

Market segment	PV	CV	TC	FE	AC	FC	ED	WT	WB	GC	HT	LT	TW	TD	HP	CW
SML	102.5	14.6	.	36.3	12.9	10.6	1312	62.2	109.4	6.4	58.7	175.1	175	14	91.2	2535
MED	106.3	16	1000	42.2	10.3	12.6	1800	62.2	109.4	6.4	57	175.6	215	17	134.9	2977
LRG	128.3	24	1815	21.6	7.4	18	2700	62.2	109.4	6.4	56.4	192.7	215	17	205.4	3252
SPT	74	8.7	.	26.3	7.3	10.6	1814	62.2	109.4	6.4	48.4	197.7	195	20	165.1	2576
SUV	130.4	40.5	1000	26.3	8	13.5	2500	62.2	109.4	6.4	72.2	181.5	215	17	191.9	3311
VAN	182	122	1600	21.2	11.5	20	2437	67.3	121.2	6.1	68.8	199.1	215	17	159	3937
TRK	52	25	2900	21.4	9.2	19	2300	57.5	122	7.6	64.9	201.2	215	15	144.5	2849

Table H.8: Utility value for Platform Scenario 4

	Model	Utility Value
AOF:		0.478
PV	SML	1
	MED	0.75
CV	MED	1
	VAN	0.256
FE	SML	0.762
AC	SPT	0.7

Appendix I: Cost Estimation

Table I.1: Cost Allocation for existing vehicle models

Market Segment	Indirect Cost ($)	Direct cost ($)	Selling cost ($)	Total Cost ($)	Price ($)
SML	3987	8461	4131	16579	18040
MED	5290	11226	5481	21996	23935
LRG	6463	13715	6697	26874	29243
SPT	6870	14579	7119	28568	31086
SUV	6566	13934	6804	27304	29711
VAN	6309	13389	6537	26235	28547
TRK	5235	11110	5425	21770	23689

Table I.2: Assumed component breakdown coefficient

Market segment	Power-train	Chassis	Body	Wheels
SML	0.35	0.3	0.25	0.1
MED	0.3	0.35	0.3	0.05
LRG	0.25	0.35	0.35	0.05
SPT	0.4	0.3	0.22	0.08
SUV	0.3	0.35	0.3	0.05
VAN	0.25	0.3	0.4	0.05
TRK	0.35	0.4	0.2	0.05

Table I.3: Direct cost allocation for existing vehicle models

Market Segment	Power-train	Chassis	Body	Wheels	Total direct cost ($)
SML	2961	2538	2115	846	8461
MED	3368	3929	3368	561	11226
LRG	3429	4800	4800	686	13715
SPT	5832	4374	3207	1166	14579
SUV	4180	4877	4180	697	13934
VAN	3347	4017	5355	669	13389
TRK	3889	4444	2222	556	11110

Table I.4: Setup activity data for learning factor identification

	Power-train	Chassis	Body	Wheels	Learning factors
Setup	0.005473	0.000666	0.001829	0.002822	0.943242
Machining	0.359335	0.162443	0.858618	0.247685	0.854054
Handling	0.010757	0.015307	0.012859	0.003753	0.849508
Maintaining	0.044514	0.023801	0.046871	0.01388	0.918849

Table I.5: Activity cost driver rate

	Setup ($/setup)	Machining ($/hours)	Handling ($/run)	Maintaining ($/labor)	Supporting ($/unit)
Power-train	100000	20000	620000	12000	88
Chassis	1000000	64000	335603	45000	264
Body	758000	14000	403405	12000	74.6
Wheels	73000	10000	75379	11000	44

Table I.6: Cost structure for Scenario 1 ($\rho *= 0.332$)

Market segment	Estimated direct cost	Estimated indirect cost	estimated selling cost	total cost	original cost	difference
SML	8027	6796	4920	19743	16579	3164
MED	10698	5563	5397	21658	21996	-339
LRG	12766	6200	6295	25261	26874	-1613
SPT	14150	6301	6787	27238	28568	-1330
SUV	13777	6706	6798	27281	27304	-23
VAN	12332	5037	5765	23134	26235	-3101
TRK	11025	4635	5197	20858	21770	-912

profit margin

Table I.7: Cost structure for Scenario 2 ($\rho *= 0.332$)

Market segment	Estimated direct cost	Estimated indirect cost	estimated selling cost	total cost	original cost	difference
SML	8067	6796	5398	20261	16579	3682
MED	12038	5598	5853	23489	21996	1493
LRG	15763	6235	7301	29299	26874	2424
SPT	12783	6301	6334	25418	28568	-3150
SUV	11880	6741	6180	24801	27304	-2504
VAN	11887	5072	5628	22587	26235	-3648
TRK	9208	4670	4606	18485	21770	-3286

profit margin

Table I.8: Cost structure for Scenario 3 (ρ ∗= 0.332)

Market segment	Estimated direct cost	Estimated indirect cost	estimated selling cost	total cost	original cost	difference
SML	8534	6400	5676	20610	16579	4031
MED	11731	5524	5727	22981	21996	985
LRG	13549	6157	6540	26246	26874	-628
SPT	10279	6219	5476	21974	28568	-6594
SUV	11980	6345	6082	24406	27304	-2898
VAN	11734	5072	5578	22383	26235	-3852
TRK	9085	4670	4565	18321	21770	-3449

*profit margin

Table I.9: Cost structure for Scenario 4 (ρ ∗= 0.332)

Market segment	Estimated direct cost	Estimated indirect cost	estimated selling cost	total cost	original cost	difference
SML	7095	6400	5677	19172	16579	2593
MED	11498	5516	5647	22661	21996	665
LRG	12582	6154	6218	24953	26874	-1921
SPT	10760	6219	5635	22614	28568	-5954
SUV	11484	6345	5917	23746	27304	-3558
VAN	12002	5673	5866	23541	26235	-2694
TRK	9050	5044	4677	18771	21770	-2999

*profit margin

Table I.10: Original direct costs

Market segment	Power-train	Chassis	Body	Wheels	Total direct cost
SML	2,961.27	2,538.23	2,115.19	846.08	8,460.76
MED	3,367.65	3,928.93	3,367.65	561.28	11,225.52
LRG	3,428.74	4,800.24	4,800.24	685.75	13,714.97
SPT	5,831.73	4,373.80	3,207.45	1,166.35	14,579.33
SUV	4,180.34	4,877.06	4,180.34	696.72	13,934.46
VAN	3,347.14	4,016.56	5,355.42	669.43	13,388.54
TRK	3,888.55	4,444.06	2,222.03	555.51	11,110.14

References

1)   Adiano, C. and Roth, A. V., 1994, "Beyond The House Of Quality: dynamic QFD", Benchmarking for Quality Management and Technology, 1994 vol. 1, pp. 25-37.

2)   Akao, Y., 1994. "Development History Of Quality Function Deployment", The Customer Driven Approach to Quality Planning and Deployment. Minato-ku, Tokyo 107 Japan: Asian Productivity Organization, 339. ISBN 92-833-1121-3.

3)   Akundi, S.V.K., Simpson, T.W., and Reed, P. M., 2005, "Multi-objective Design Optimization For Product Platform and Product Family Design Using Genetic Algorithm," ASME 2005 International Design Engineering Technical Conferences & Computers and Information in Engineering Conferences, DETC2005-84905

4)   Alizon, F, Shooter, S.B., and Thevenot, H.J., 2006 "Design Structure Matrix Flow For Improving Identification And Specification Of Modules", Proceedings of the ASME Design Engineering Technical Conference, Proceedings of 2006 ASME International Design Engineering Technical Conferences and Computers and Information In Engineering Conference, DETC2006, 2006, 13p.

5)   Allada, V., & Jiang, L., "New Modules Launch Planning For Evolving Modular Product Families", ASME Design Engineering Technology Conf., 2002, Paper No. DETC20020DFM-34190.

6)   Baldwin, C. Y. and Clark, K. B., 1997, "Managing In The Age of Modularity," Harvard Business Review, vol. 75, pp. 84-93, 1997.

7)   Bell, D. E., Raiffa, H., & Tversky, A. (Eds.). (1988). Decision Making: Descriptive, Normative, And Prescriptive Interactions. New York: Cambridge University Press, ISBN: 0521368510. (Richard Gonzalez, Psychology Department)

8)   Ben-Arieh, D. and Qian, L., 2003, "Activity-based Cost Management For Design and Development Stage," Int. J. Prod. Econ., 2003, 83, 169–183.

9)   Box, G.E.P.; Hunter, W.G.; Hunter, J.S, 1978, "Statistics for experimenters: an introductory to design data analysis and model building", Wiley, New York (USA)

10)   Bremmer, R., 1999, "Cutting-Edge Platforms", Financial Times Automotive World, September, P30-38

11)   Bremmer, R., 2000, "Big, Bigger, Biggest", Automotive World, June, p36-44

12)   Browning, T.R, 2001, "Applying The Design Structure Matrix To System Decomposition And Integration Problem: A Review and New Directions", IEEE Transactions on Engineering Management, Vol 48, No. 3, August 2001

13)   Caffrey, R.T., Simpson T.W., Henderson R., and Crawley E., 2002, "The Technical Issues With Implementing Open Avionics Platforms For Spacecraft", the 40th AIAA Aerospace Science Meeting and Exhibit, Reno, NV, AIAA, AIAA-2002-0319

14)   Cagan, J. and Vogel, C. M., 2002, "Creating Breakthrough Products: Innovation From Product Planning To Program Approval," Upper Saddle River, NJ: Prentice Hall PTR, 2002.

15)   Carney, D., 2004, "Platform Flexibility", Automotive Engineering International, pages147-149, February 2004.

16)   Cavalieri, S., Maccarrone, P., and Pinto, R., 2004, "Parametric Vs Neural Network Models For The Estimation Of Production Costs: A Case Study in the Automotive Industry," Int. J. Prod. Econ., 91_2_, pp. 165–177.

17)   Choubey, A.M., 2007, "Platform Based Approach For Economic Production Of A Product Family", Master Thesis, Department of Industrial & Manufacturing Systems Engineering, Kansas State University Manhattan.

18)   Cook, K., "Product Management: Value, Quality, Cost, Price, Profit And Organization", Chapman & Hall, 1997.

19)   Cooper, R. and Kaplan, R.S., 1991, "Profit Priorities From Activity-based Costing", Harv. Busi. Rev., 69, 130–135.

20)   Cusumano, M.A., and Nobeoka, K., 1998, "Thinking Beyond Lean," The Free Press, New York, NY

21) Dahmus, J. , Gonzalez-Zugasti, J. and Otto, K., 2001, "Modular Product Architecture," Design Studies, vol. 22, pp. 409-424, 2001.

22) Dai, Z. and Scott, M.J, 2002, "Effective Product Family Design Using Preference Aggregation," Journal of Mechanical Design, Vol. 128, Issue 4

23) Dai, Z. and Scott, M.J., 2007, "Product Platform Design Through sensitivity Analysis And Cluster Analysis", Journal of Intelligent Manufacturing 18:97-113, 2007

24) D'Souza, B. and Simpson' T.W., 2003, "A Genetic Algorithm Based Method For Product Family Design Optimization," Engineering Optimization, 34(1):1-18

25) Ericsson, A. and Erixon, G., 1999, "Modular Product platforms," ASME Press, 1999, p29-41.

26) Fellini, R., Mokkolaras, M., Michelena, N., Papalambros, P., Perez-Durate, A., Saiton, K., and Fenyes, P., 2004, "A Sensitivity Based Commonality Strategy For Family Products Of Mild Variation, With Application To Automotive Body Structures," Struct Multidisc Optim 27. 89-96, 2004

27) Fernandez, M.G., Seepersand, C.C., Rosen, D.W., Allen, J.K., and Mistree, F., 2005, "Decision Support In Concurrent Engineering-The Utility based Selection Decision Support Problem," Concurrent Engineering: Research and Appication, 13-1, 2005

28) Ficko, M., Drstvensek, I., Brezocnik, M., Balic, J., and Vaupotic, B., 2005, "Prediction Of Total Manufacturing Costs For Stamping Tool On The Basis Of CAD-Model Of Finished Product," J. Mater. Process. Technol., 164–165, pp.1327–1335

29) Fine C.H., 2005, "Are You Modular Or Integral? Be Sure Your Supply Chain Knows", Strategy+Business, Issue 39, Summer 2005

30) Fixson, S.K., 2001, "Three Perspectives On Modularity-A Literature Review Of A Product Concept For Assembled Hardware Products," MIT, Boston, MA, Working Paper ESD-WP-2001-06, October 2001

31) Fujita Y.K, 2000, "Product Variety Optimization Under Modular Architecture," Proceedings of Third International Symposium on Tools and Methods of Competitive Engineering (TMCE2000), pp.451-464

32) Fujita, Y. K., Akagi, S., Yoneda'\, T. and Ishikawa, M., 1998, "Simultaneous Optimization Of Product Family Sharing System Structure And Configuration," ASME Design engineering Technical Conferences-Design for Manufacturing, Las Vegas, NV, ASME, DETC99/DFM-8923

33) Fujita, Y. K., Sakaguchi, H. and Akagi, S., "Product Variety Development And Its optimization Under Modular Architecture and Module Commonalization," Proceedings of the 1999 ASME Design Engineering Technical Conferences

34) Fujita, Y. K., Takagi, and Nakayama, 2003, "Assessment Method Of Value Distribution For Product Family Deployment", International Conference on Engineering Design, ICED, Stockholm, Aug.19-21, 2003.

35) Fujita, Y.K., and Yoshida, H., 2004, "Product Variety Optimization Simultaneously Designing Module Combination And Module Attributes, " Concurrent Engineering: Research and Application, Vol. 12(2):105-118 Paper No. DETC2001/DAC-21058.

36) Fung, Y.K., Chen, Y. and Tang, J., "A Quality-engineering-based Approach For Conceptual Product Design," International Journal of Advanced Manufacturing Technology (2007), 32:1064-1073

37) Fung, Y. K. Law, D. S. T. and Ip, W. H., 1999, "Design Targets Determination For Inter-dependent Product Attributes In QFD Using Fuzzy Interference," Integrated Manufacturing Systems, vol. 10, pp. 376-383, 1999.

38) Fung, Y.K., Tang, J., Tu, Y. and Chen, Y., 2003, "Modelling Of Quality Function Deployment Planning With Resource Allocation", Res. Eng. Design, 2003, 14(4), 247–255.

39) Gershenson, J. K., Prasad, G. P. and Zhang, Y., 2004, "Product Modularity: Measures And Design Methods," Journal of Engineering Design, vol. 15, pp. 33-55, February 2004.

40) Gonzalez-Zugasti, J. P. and Otto, K. N. 2000. "Platform-Based Spacecraft Design: A Formulation And Implementation Procedure." 2000 IEEE Aerospace Conference, Big Sky, Montana

41)   Gonzalez-Zugasti, J.P., Otto, K.N., 2001, and Baker, J. D., "Assessing Value In Platformed Product Family Design," Research on Engineering Design 13 (2001), 30-41

42)   Gonzales-Zugasti J.P., Otto, K.N., 2000, "Modular-Based Platform Design", Proceedings Of DETC'00 ASME Design Engineering Technical Conferences and Computers and Information in Engineering Conference Baltimore, Maryland, September 10-13, 2000

43)   Guyot, N.E., 1996, "Fuzzy Logic And Utility Function For Multiobjective Optimization Of Automotive Joints," Master Thesis, Virginia Tech.

44)   Hajare, A. D., 1998, "Parametric Costing—Steel Wire Mill," Proceedings of the Annual Convention of the Wire Association International, pp. 172–178.

45)   Hartigan, J. A. and Wong M. A., 1979. "A K-Means Clustering Algorithm". Applied Statistics 28 (1): 100–108

46)   Hilton, R.W., Maher, M.W. and Selto, F.H., 2000, "Cost Management: Strategies For Business Decisions," 2000 ,Irwin/McGraw-Hill: Boston, MA.

47)   Hundal, M.S., 1997, "Product Costing: A Comparison Of Conventional And Activity-based Costing Methods," J. Eng. Des., 1997, 8, 91–103.

48)   Hundal, M.S., 1993, "Design to Cost," Concurrent Engineering: Contemporary Issues And Modem Design Tools, H. R. Parsaei and W. G. Sullivan, eds., Chapman and Hall, London, pp. 330–351.

49)   Jiao, J. and Zhang, Y., "Product Portfolio Planning With Customer-engineering Interaction", IIE Transactions, 2005, 37 (9), pp. 801-814.

50)   Kalligeros, K., Weck, O. and de Neufville, R., 2006, "Platform Identification Using Design Structure Matrices", Sixteenth Annual International Symposium of the International Council on System Engineering (INCOSE), July 8-14, 2006.

51)   Keeney, R. L., & Raiffa, H. (1993), "Decisions With Multiple Objectives: Preferences And Value Tradeoffs," New York: Cambridge University Press, ISBN: 0521438837. (Richard Gonzalez, Psychology Department)Kim, K.J. Moskowitz, H.,

Dhingra, A. and Evans, G., 2000, "Fuzzy mutlicriteria models for quality function deployment," European Journal of Operational Research, vol. 121, 2000.

52)   Kreng, V. B. and Lee, T.P., 2004, "QFD-based Modular Product Design With Linear Integer Programming-A Case Study," Journal of Engineering Design, vol. 15, pp. 261-284, June 2004.

53)   Krishnan, V. and Ulrich K., 2001, "Product Development Decisions: A Review Of The Literature," Management Science, vol. 47, pp. 1-21, January 2001.

54)   Kulkarni, R.S., Allen, J.K., Mistree, F., 2005, "Designing Product Families For A Changing Market Space", Proceedings of the ASME International Design Engineering Technical Conferences and Computers and Information in Engineering Conference - DETC2005, v 2 B, Proceedings of the ASME International Design Engineering Technical Conferences and Computers and Information in Engineering Conferences - DETC2005: 31st Design Automation Conference, 2005, p 1029-1051

55)   Kumar, R. and Allada, V., 2005, "Customer Need Driven Function Behavior Platform Formation", 2005 ASME Design Engineering Technical Conferences & Computer and Information in Engineering Conference, Sep. 24-28, 2005, Long Beach, California, USA

56)   Kurtadikar, R.M. et al. ,"A Customer Needs Motivated Conceptual Design Methodology For Product Portfolios," Proceeding of DETC'04 ASME 2004 Design engineering Technical Conferences and Computers and Information in Engineering Conferences, DETC2004-57289

57)   Lehnerd, A.P., 1987, "Revitalizing The Manufacture And Design Of Mature Global Products", Technology and Global Industry: Companies and Nations in the World Economy, B.R.Guile and H.Brooks, eds, National Academy Press, Washington D.C., p49-64

58)   Lewis, J., 2000, "Metrics Mapping Cuts Estimating Time," Des. News, 55_18_, pp. 107–110.

59) Li, H. and Azarm, S., "An Approach For Product Line Design Selection Under Uncertainty And Competition", ASME Journal of Mechanical Design, 2002, 124(3), pp. 385–392.

60) Li, Y.F., Hu, N., Terpenny, J. 2009, "An Approach To Address Independence Of Criteria In Multi-attribute Decision-Making", Proceedings of the 2009 Industrial Engineering Research Conference

61) Lin, M.C, Wang, C.C., and Chen, T.C., "A Strategy For Managing Customer-Oriented Product Design," Current Engineering: Research and Application, Vol. 14-3, Sep. 2006.

62) Liu G.P., Yang J.B., and Whidborne J.F., 2002, "Multiobjective Optimization And Control," Research Studies Press LTD., Baldock, Hertfordshire, England

63) Marsot, J., 2005, "QFD: A Methodological Tools For Integration Of Ergonomics At The Design Stage", Applied Ergonomics, vol. 36, pp. 185-192, 2005.

64) Martin, M. and Ishii, K., 2000, "Design For Variety: A Methodology For Developing Product Platform Architectures," presented at ASME International Design Engineering Technical Conferences & Computers and Information In Engineering Conference, Baltimore, Maryland, USA, 2000. [DETC2000/DFM-14021]

65) Martin, M., and Ishii, K., 2002, "Design For Variety: Developing Standardized And Modularized Product Platform Architectures", Research in Engineering Design, 2002, 13:213-235.

66) Messac, A., Martinez, M.P., and Simpson, T.W., 2002a, "Effective Product Family Design Using Physical Programming," Engineering Optimization," 34(3):245-261

67) Messac, A., Martinez, M.P., and Simpson, T.W., 2002b, "A Penalty Function For Product Family Design Using Physical Programming," ASME Journal of Mechanical Design, 124(2):164-172

68) Messac, A., Martinez, M.P, and Simpson, T.W., 2002, "Introduction Of A Product Family Penalty Function Using Physical Programming", Transaction of the ASME, Vol. 124, June 2002

69)   Meyer, M. H. and Lehnerd, A. P., 1997, "The Power Of Product Platforms: Building Value And Cost Leadership", New York, NY: The Free Press, 1997.

70)   Moore, et al., 1999, "Using Conjoint Analysis To Help Design Product Platforms," Journal of Product Innovation Management, 1999, 16:27-39

71)   Muffatto, M., 1999, "Introducing A Platform Strategy In Product Development", International Journal of Production economics, 60-61:145-153

72)   Nayak, R.U., Chen, W. and Simpson, T.W., 2002, "A Variation-Based Method For Product Family Design", Engineering Optimization," 2002, 34(1):65-81

73)   Nelson, S. A.II, Parkinson, M.B., and Papalambros, P.Y., 2001, "Multicriteria Optimization In Product Platform Design," ASME Journal of Mechanical Design, 123(2): 199-204

74)   Niazi, A., Dai, J. S., Balabani, S., Seneviratne L., 2006, "Product Cost Estimation: Technique Classification And Methodology Review," Journal of Manufacturing Science and Engineering, Vol. 128, 563-575

75)   Ortega, R. A., Kalyan-Seshu, U., and Bras, B., "A Decision Support Model For The Lifecycle Design Of A Family Of Oil Filters," ASME Design Engineering Technical Conference, 1999, Paper No. DETC/DAC-8612.

76)   Otto, K. and Wood, K., 2001, "Product Design: Techniques In Reverse Engineering And New Product Development," New York: Prentice Hall, 2001.

77)   Otto K. and Holtta, K., 2004, "A Multi-criteria Framework For Screening Preliminary Product Platform Concepts," presented at ASME International Design Engineering Technical Conferences & Computers and Information In Engineering Conference, Salt Lake City, UT USA, 2004. [DETC2004-57256]

78)   Pahl, G., and Beitz, W., 1996, "Engineering Design: A Systematic Approach," 2nd ed., Springer-Verlag, Berlin.

79)   Park J., and Simpson T.W., 2005, "Development of a production cost estimation framework to support product family design," International Journal of Production Research, Vol. 43, No., 4, 731-772

80) Pimmler, T.U. and Eppinger, S.D., 1994, "Integration Analysis Of Product Decompositions", ASME Design Engineering Technical Conferences - Design Theory and Methodology, Minneapolis, MN, ASME.

81) Pine, B.J. 1993, "Mass Customization: The New Frontier In Business Competition", Harvard Business School Press, Boston, MA.

82) Poli, C., Escudero, J., and Fermandez, R., 1988, "How Part Design Affects Injection-Moulding Tool Costs," Mach. Des., 60_1_, pp. 101–104.

83) Rai, R., and Allada, V., 2003, "Modular Product Family Design: Agent-based Pareto-optimization And Quality Loss Function-based Post-optimal Analysis," International Journal of Production Research, 41(17): 4075-4095

84) Raiffa, H. and Keendy, R., 1993, "Decision With Multiple Objectives: Preferences And Value Tradoffs," Wiley, New York, NY

85) Rehman, S., and Guenov, M. D., 1998, "A Methodology For Modeling Manufacturing Costs At Conceptual Design," Comput. Ind. Eng., 35_3–4_, pp. 623– 626.

86) Reich,Y. and Levy, E., 2004, "Managing Product Design Quality Under Resource Constraints", Int. J. Prod. Res., 2004, 42(13),2555–2572.

87) Robertson, D. and Ulrich, K., 1998, "Planning For Product Platforms," Sloan Management Review", ABI/INFORM Global ,39(4),19-31

88) Romesburg, H. C., 1990, "Cluster Analysis For Researchers," Robert E. Krieger Publishing Company, Malabar, Florida.

89) Rothwell, R. and Gardiner, P., 1990, "Robustness And Product Design Families In Design Management: A Handbook of Issues and Methods", Oakley, M. (Ed.), Basil Blackwell Inc., Cambridge, MA, p 279-292

90) Sahin, A., 2007, "A Data Clustering Approach To Support Modular Product Family Design", Ph.D Dissertation, Department of Industrial Systems and Engineering, Virginia Polytech, 2007.

91) Sahinidis N.V., 2000, "Branch And Reduce Optimization Navigator", University of Illinois

92)   Sanderson, S.W. and Uzumeri, M., 1997, "Managing Product Families," Irwin, Chicago, IL

93)   Sabbagh, K., 1996, "Twenty First Century Jet-The Making and Marketing of The Boeing 777", Scribner, New York, NY

94)   Satty, T., 1980, "The Analytic Hierarchy Process," McGraw-Hill, New York, NY

95)   Sawaragi, Y., Nakayama, H. and Tanino, T. (1985)."Theory Of Multiobjective Optimization," (vol. 176 of Mathematics in Science and Engineering). Orlando, FL: Academic Press Inc.

96)   Seepersad, C. C., Hernandez G., Allen J. K., 2000, "A Quantitative Approach To Determining Product Platform Extent",Proceedings of DETC00/DAC, 2000 ASME Design Engineering Technical Conference, DETC 2000/DAC-14288, September 10-13, Baltimore, Maryland, 2000

97)   Seepersad, C. C., F. Mistree and J. K. Allen, 2002, "A Quantitative Approach For Designing Multiple Product Platforms For An Evolving Portfolio of Products," ASME Advances in Design Automation, Montreal, Canada. Paper Number: DETC2002/DAC-34096.

98)   Siddique Z., 2005, "Product Family Redesign Using A Platform Approach" in Product Platform and Product Family Design: Methods and Applications, T.W. Simpson, Z. Siddique, and J. R. Jiao, Eds. New York: Springer, 2005, pp. 359-376

99)   Siddique, Z., and Repphun B., 2001, "Estimating Cost Savings When Implementing A Product Platform Approach," Concurrent Engineering: Research and Applications, Vol. 8, No. 4

100)   Shirley, G. V., 1990, "Models For Managing The Redesign And Manufacture Of Product Sets," Journal of Manufacturing and Operations Management, Vol. 3, pp. 85-104.

101)   Simpson, T.W., 2005, "Methods For Optimizing Product Platforms And Product Families" in Product Platform and Product Family Design: Methods and Applications, T.W. Simpson, Z. Siddique, and J. R. Jiao, Eds. New York: Springer, 2005, pp. 142-156

102)   Simpson, T.W., 2004, "Product Platform Design And Customization: Status And Promise," Artificial Intelligence for Engineering Design, Analysis and Manufacturing, 18(1), p3-20

103)   Simpson, T. W. and D'Souza, B., "A Variable Length Genetic Algorithm For Product Platform Design", 9th AIAA/ISSMO Symposium on Multidisciplinary Analysis and Optimization, 2002, Atlanta, GA, September 4-6, AIAA, AIAA-2002-5427.

104)   Simpson, T.W. and D'Souza, B., 2004, "Assessing Variable Levels Of Platform Commonality Within A Product Family Using A Multi-objective Genetic Algorithm," Concurrent Engineering: Research and Application, 12(2):119-130

105)   Simpson, T.W., Maier, J. R. A., and Mistree, F., 2001, "Product Platform Design: Method And Application," Research in Engineering Design, 13 (1): 2-22

106)   Simpson, T.W., Marion, O., Weck, O., Holltta-Otto, K., Kokkolaras, M., and Shooter, S.B., 2006, "Platform-based Design And Development: Current Trends And Needs In Industry," ASME 2006 International Design Engineering Technical Conferences & Computers and Information in Engineering Conferences, DETC2006-99229

107)   Stone, R. B., Wood, K. L. and Crawford, R. H., 2000, "A Heuristic Method For Identifying Modules For Product Architectures," Design Studies, vol. 21, pp. 5-31, 2000.

108)   Stadzisz, P. C. and Henrioud, J. M., 1995, "Integrated Design Of Product Families And Assembly Systems," IEEE International Conference on Robotics and Automation, Nagoya, Japan, IEEE, Vol. 2, pp. 1290-1295.

109)   Strong, M. B., Magleby, S. P. and Parkinsonn, A. R., 2003, "A Classification Method To Compare Modular Product Concepts," presented at ASME International Design Engineering Technical Conferences & Computers and Information In Engineering Conference, September 2-6, 2003. [DETC2003/DTM-48661]

110)   Suh, E.S., 2005, "Flexible Product Platforms", Ph.D dissertation, MIT, Sep., 2005

111)   Sydenham, P.H., 2004, "Systems Approach To Engineering Design", Artech House Inc., 2004

112)   Thevenot, H.J. and Simpson, T.W., 2004, "A Comparison Of Commonality Indices For Product Family Design," Proceedings of the 2004 ASME Design Engineering Technical Conferences - 16th International Conference on Design Theory and Methodology, September, Salt Lake City, UT.

113)   Tseng, M. and Jiao, J., 1998, "Design For Mass Customization By Developing Product Family Architecture," presented at ASME International Design Engineering Technical Conferences & Computers and Information In Engineering Conference, Atlanta, GA USA, September 13-16, 1998. [DETC98/DFM-5717]

114)   Ulrich K., 1995, "The Role Of Product Architecture In The Manufacturing Firm", Research Policy 24, 419-440

115)   Ulrich K. and Eppinger, S., 1995, "Product Design And Development," New York: McGraw-Hill, 1995.

116)   Vyas, A., Santini D. and Cuenca R., 2000, "Comparison of indirect cost multipliers for vehicle manufacturing", http://msl1.mit.edu/classes/esd123/vyas.pdf

117)   Ward, Joe H. (1963). "Hierarchical Grouping to Optimize an Objective Function". Journal of the American Statistical Association 58 (301): 236–244.

118)   Weck. O., 2006, "Determining Product Platform Extent", in Product Platform and Product Family Design: Methods and Applications, T.W. Simpson, Z. Siddique, and J. R. Jiao, Eds. New York: Springer, 2005, pp. 241-301

119)   Weck, M., Klocke, F., Schell, H., and Ruenauver, E., 1997, "Evaluating Alternative Production Cycles Using The Extended Fuzzy AHP Method," European Journal of Operational Research 100 (1997) 351-366

120)   Weck, O., Suh, E.S. and Chang, D., 2003, "Product Family And Platform Portfolio Optimization", Preceedings of DETC03, ASME Design Engineering Technical Conferences, Sep. 2-6, 2003, Chicago, Illinois, USA.

121)   Wilhelm, B., 1997, "Platform And Modular Concepts At Volkwagen-Their Effect On The Assembly Process," Transforming Automobile Assembly: Experience in Automation and Work Organization, Springer, New York, p146-156

122) Williams, C.B., Allen, J.K., Rosen, D.W. and Mistree, F, 2004, "Designing Platforms For Customizable Products In Markets With Non-uniform Demand", Proceedings of the ASME Design Engineering Technical Conference, v 3, Proc. of the ASME Des. Eng. Tech. Conf. and Comput. and Inf. in Eng. Conf. 2004: Vol. 3: 16th Int. Conf. on Des. Theory and Methodol.: 2nd Symp. on Int. Issues in Eng. Des.: Integr. of Materials Micro, 2004, p 337-349

123) Williams, C.B., Allen, J.K. , Rosen, D.W. and Mistree, F., 2007, "Designing Platforms For Customizable Products And Processes In Markets Of Non-uniform Demand", Concurrent Engineering: Research and Applications, v 15 n 2, June 2007, 201-216

124) Winterfelt, V. and Edwards, W., "Decision Analysis And Behavioral Research," Cambridge University Press, 1986.

125) Xu, l., Li,z., Li, S., and Tang, F., 2004, "A Decision Support System For Product Design In Concurrent Engineering," Decision Support System 42 (2004), 2029-2042

126) Yano, C. and Dobson, G., "Profit Optimizing Product Line Design, Selection And Pricing With Manufacturing Cost Considerations", Product Variety Management: Research Advances, Ho, T.H. and Tang, C.S. (eds.), Kluwer Academic Publishers, Boston, MA, 1998, pp. 145–176.

127) Yassine, A.A., and Falkenburg, D.R., 1999, "A Framework For Design Process Specifications Management", Journal of Engineering Design, 1999, 10(3): 223-234

128) Ye, X., Gershenson, J. K., Khadke, K., Lai, X. and Guo, F., 2005, "An Introduction To product Family Evaluation Graphs," presented at ASME International Design Engineering Technical Conferences & Computers and Information In Engineering Conference, Long Beach, California, 2005. [DETC2005-85229]

129) Yu, J.S., Gonzalez-Zugasti J.P., Otto K.N., 1998, "Product Architecture Definition Based Upon Customer Demands", Proc. 1998 ASME Design Engineering Technical Conferences, Atlanta, USA, 1998.

130) Zamirowski, E.J. and Otto, K.N., 1999, "Identifying Product Portfolio Architecture Modularity Using Function And Variety Heuristics", Proceedings of 11th

International Conference On Design Theory and Methodology, 1999 ASME Design Engineering Technical Conferences, Sep. 12-15, 1999, Las Vegas, Nevada

131) Zha, Xf, Sriram, R.D., and Lu, W.F., 2004, "Evaluation And Selection In Product Design For Mass Customization: A Knowledge Decision Support Approach", Artificial Intelligence for Engineering Design, Analysis and Manufacturing, 18,87-109