ABSTRACT

YOUNG, KEVIN MATTHEW. Informed Genetic Algorithm Crossover Operators for Market-Driven Design.

Heuristic optimization algorithms have seen increased adoption as a means of finding solutions to complex and large-scale problems within the design community. Previous research looked into the promising results associated with tuning genetic algorithms when applied to solving product line optimization problems (i.e. computational benefits and improvements in solution quality). This study investigates the effects of developing crossover operators influenced by heuristic rules for product line optimization problems, specifically in regards to optimizing the market share of preference and profit of both an MP3 product line and an automobile product line. Informed crossover operators are developed and explored that use problem-related information to inform their behavior. Of the crossover techniques studied in this work, a crossover method that retained the product with the highest relative market share of preference within each product line was found to be most effective. The presented results indicate a significant improvement in computational efficiency and algorithm effectiveness when compared to standard genetic algorithm tuning methods. Future work in this subject will investigate the development of informed selection and mutation operators, as well as problem informed schema.

Informed Genetic Algorithm Crossover Operators for Market Driven Design

by
Kevin Matthew Young

A thesis submitted to the Graduate Faculty of
North Carolina State University
in partial fulfillment of the
requirements for the degree of
Master of Science

Mechanical Engineering

Raleigh, North Carolina

2016

APPROVED BY:

Dr. Gregory Buckner

Dr. Mark Pankow

Dr. Scott Ferguson
Committee Chair

DEDICATION

Dedicated to my fiancée Kimberly Schreiber, my parents Paul and Christina Young, and my brother Connor Young, for their endless hours of support throughout my educational career.

BIOGRAPHY

Kevin Young completed both his Bachelor of Science degree in Mechanical Engineering and Bachelor of Science degree in Mathematics at North Carolina State University in May 2014. Upon graduation, he re-enrolled at North Carolina State University to pursue a Master of Science degree in Mechanical Engineering in the fall of 2014. Kevin joined the System Design Optimization Lab in the summer of 2015 and began working towards the completion of his thesis under the direction of Dr. Scott Ferguson.

ACKNOWLEDGMENTS

The author expresses his appreciation for his advisor, Dr. Scott Ferguson, who has supported him throughout his entire graduate career at NC State and who has gone above and beyond the usual role of an advisor to assist in preparing the author for his future. The author would also like to thank his lab-mates Krista Edwards, Charlotte Lawrence, Daniel Long, Jaekwan Shin, and Samantha White for their assistance and support during his tenure in the System Design Optimization Lab.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1 – INTRODUCTION

1.1 Introduction

As consumers demand products that are more competitively configured and priced, companies have begun to leverage market research advancements that allow for a more comprehensive understanding of consumer preferences at an individual level [1–7]. Exploring the heterogeneous consumer preferences that are estimated by these market research models has revealed untouched market segments and demonstrated the missed business opportunities associated with a single-product offering [8,9]. To maximize market competitiveness, companies have established product line offerings that fill a wide variety of consumer needs while balancing business objectives associated with maximizing market share and profit. However, formulating product line design problems can lead to large, complicated design spaces that many optimization algorithms struggle to effectively navigate [10]. Companies must understand the delicate interaction between product line size and resulting product configurations, as the risks associated with poorly designed product lines can be great (i.e. product cannibalization, missed market capture, reduction of revenue, etc.).

1.2 Product Line Design

A product line is defined as “the set of related products that are offered by a single company” [11]. Product line design calls for the simultaneous development of multiple related product variants and abandons the commonly pursued strategy of designing “one product at a time,” allowing a company the luxury of expanding their product lines in simultaneous fashion to

respond to the ever-changing needs of a heterogeneous market [12]. When firms pursue efforts to alter products one at a time, many issues can arise. For example, a firm can update a product variant that seeks to maximize its respective market share. However, by capturing the market share associated with said product, the firm may inadvertently be diverting consumers in the market away from other products that the company offers, a process known as product cannibalization [13]. By developing product lines that contain products evolving to market changes together, companies can ensure that niche markets are identified and filled while preventing cannibalization.

1.3 Market-Based Product Design

With the ever-increasing computational resources available to engineers and designers, consumer preferences can be captured in increasingly more accurate ways and used to maximize objectives that firms consider important, the most common of which are market share and/or profit. One technique used to capture the information needed to estimate customer preferences is a choice-based conjoint (CBC) survey [14]. These consumer preferences can then be estimated in the form of part-worth coefficients, or values that dictate a respondent's preference towards (or aversion to) a certain attribute level [15–18].

As the number of part-worth estimates increases (as is common with most engineering design problems), the number of product line design possibilities increases exponentially [19,20]. It is often not reasonable to expect designers to be able to infer a product configuration that potential customers will be drawn to. Rather, optimization algorithms can be used to

determine optimal solutions by using part-worth estimates and maximizing an objective by designing around these preferences [21]. However, many optimization algorithms cannot handle problems of such magnitude and can fail to reach feasible solutions that improve upon the objectives being sought after.

1.4 Product Line Optimization

A commonly made assumption with product line design is that a product can be decomposed into product features, and that those features can be described in discretized levels [22]. The design variables for a product line optimization then become the feature levels used in each product being designed. In addition to feature levels, previous research has established the use of pricing as a design variable, increasing the size of the design string and often making the problem mixed-integer [23–25].

Large product line design problems were initially considered to have more than 20 binary variables [26], but others have expanded on this definition to include product lines of 6 – 8 products with 20 – 24 variables per variant, resulting in problems with at least 1.8E+33 possible configurations [27]. To highlight the computational challenges posed by these problems, Belloni et al. [28] introduced a problem with approximately 5E+15 feasible product line solutions. Solving this problem using complete enumeration would take over 5,000 years, and over 1 week using a branch-and-bound algorithm running at a rate of 30,000 evaluations per second. As product development times shorten [29], Belloni et al. [28] further explain that most firms would consider one week of computation to be an upper limit of

acceptable time. If limited to a single day of computation, only 8.6E-7 percent of the total design space could be explored.

The complexity associated with product line configuration problems has led the engineering design community to focus research efforts on the development of heuristic optimization techniques to determine solutions and/or solution spaces for these problems [21,30–34]. These heuristic methods have proven to produce more complete and thorough solutions spaces when compared with greedy and rule-based approaches.

1.5 Genetic Algorithms

For the purposes of this research, a genetic algorithm (GA) is used to optimize product line configurations. Developed in 1975 by John Holland, GAs are heuristic algorithms based on Charles Darwin's theory of natural selection. Ultimately, the basis of the algorithm is to promote a "survival of the fittest" directive to guide the search of complex design spaces [35]. A GA begins by initializing a "population" of possible designs, at which point a selection process is initiated to determine the strongest (i.e. best performing) design strings, which will be selected for "reproduction" (referred to as "parents"). These selected parents undergo a reproductive process referred to as crossover, where information from the two parental design strings are interchanged and "children", or design strings made up of the information from the two parents) are generated and re-introduced into the population. Populations are then trimmed down via a culling procedure to maintain population size, at

which point a full “generation” cycle is completed. The process then repeats until a convergence of the population occurs.

Research has revealed the significant advantages offered by GAs. First and foremost, GAs are highly receptive to parallel processing, as design strings can be evaluated by multiple computers/processors in parallel [35]. Another key feature of a GA is that it is highly tunable. Each operator is extremely customizable, and the research conducted in this thesis leverages this aspect of a GA in order to achieve computational benefits and improved solution quality. Finally, the heuristic nature of a GA allows it to be extremely effective in complex solutions spaces, allowing for the discovery of optimal solutions that analytical approaches may not find [35]. It is for these aforementioned reasons that a genetic algorithm is used in this research.

1.6 Research Questions

Previous research conducted on the topic of product line optimization has explored the implementation of problem-specific data in the tuning process of a GA, specifically in regards to tuning initial populations (referred to as targeted populations). This work, focusing on initializing populations for use in a genetic algorithm, implemented a sub-optimization algorithm that determined optimal products at the respondent level to seed a population.

The research presented in this thesis extends the concept of problem-data informed operators for use in a genetic algorithm used to optimize product line configurations. However, the proposed research will implement problem-specific data at a population level, utilizing share

of preference data (that is dependent upon a population as opposed to a single respondent) to inform crossover methods. The implementation of this population-scale data within an operator where a majority of design string alterations occurs (the crossover operator) will theoretically yield stronger benefits when applied to complex product line optimization problems. Figure 1.1 details a proposed method for influencing an informed crossover using market share of preference data, and the method presented will guide the development of the proposed crossover method. The share of preference data in question will be used to inform the crossover methods so as to differentiate them from standard crossover methods that make design string changes based purely on random choices (i.e. a random uniform change, a random selection over a distribution, etc.)

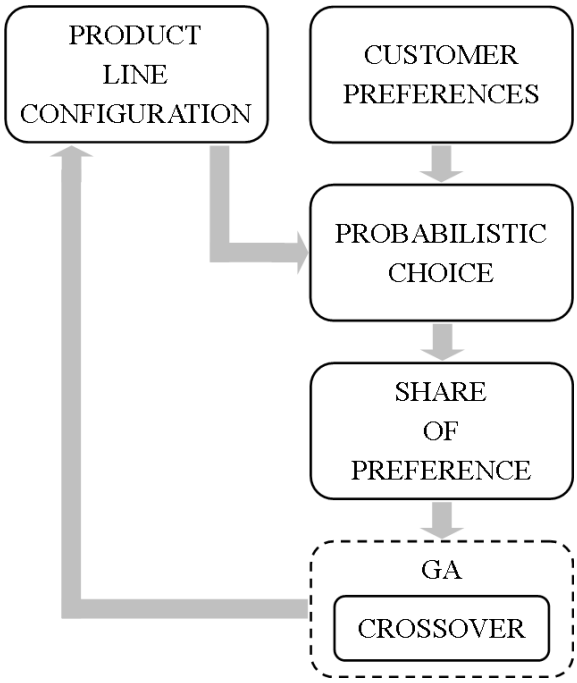

Figure 1.1: Process Overview of Proposed Research

The work in this thesis addresses two research questions concerning the implementation of problem-specific data within algorithms aimed towards optimizing a product line configuration. The first research question seeks to identify the advantages offered through the implementation of this data, and the second research question highlights how problem formulation affects the advantages offered by this proposed implementation.

Research Question 1: What are the advantages of using problem-specific information within the crossover operator when optimizing product line configurations?

Research Question 1 (RQ1) is intended to explore the effects of using problem specific information to develop more effective crossover operators for optimizing product line configuration. These crossover operators (referred to henceforth as informed operators) will be developed by leveraging information that is specifically related to product line configuration, namely market share and profit related information. A crossover operator was selected for this study because of its strong influence on updating the population of a GA. After developing a range of informed crossover operators in Chapter 3, a full factorial experimental design composed of varying problems, product line sizes, initialization techniques, and the crossover operators is used to fully explore the benefits attributed to the informed crossover operators. These benefits will be leveraged against a standard crossover method commonly used within genetic algorithms.

Research Question 2: How does problem formulation impact the performance of crossover operators that use problem-specific data?

As previously mentioned, the effectiveness of the informed crossover operators will be tested using a full factorial experimental design. Following analysis of the advantages offered by these informed crossover operators when compared against a standard crossover method, an in-depth analysis of the advantages and disadvantages offered between the developed informed crossover methods is conducted. The various informed crossover methods will be compared and recommendations on the implementation of said operators will be made based on the data collected through test trials.

1.7 Chapter Summary

Chapter 2 provides background information that describes the foundational components of market-driven product line optimization. The experimental method used to collect data is presented in Chapter 3, and also contains information on the development of the informed crossover methods. Chapters 4 and 5 introduce two single objective test problems used to determine the advantages offered by the informed crossover methods. Chapter 6 then presents a multi-objective approach to the problem given in Chapter 5. Lastly, Chapter 7 presents a summary of the findings of this research, answering RQ1 and RQ2 and presenting potential avenues for future work.

CHAPTER 2 – BACKGROUND

2.1 Modeling Customer Preferences

A significant challenge of engineering design is defining product line configuration and price in a way that does not unintentionally sacrifice profit for market share (or visa-versa, or both) [9]. Yet, polling a large customer base about their preference towards every possible product configuration is impractical. This has led to the development of customer preference models that allow designers to gauge preferences from a much smaller data source without sacrificing knowledge about market heterogeneity [2,5,36]. Shiau and Michalek [37] note that without models that accurately capture market heterogeneity, product line optimizations often yield suboptimal results with significant overestimation of expected performance. The research presented in this thesis uses information gained from heterogeneous customer preference estimates to discover effective product line configurations in a more informed way.

A common survey approach to capture the information needed to estimate customer preference is a choice-based conjoint (CBC) analysis. Respondents are presented with multiple questions, called choice tasks, and asked to choose the product that they prefer from a set of hypothetical product configurations [38,39]. CBC analysis has gained traction in recent years because it more realistically models a consumer's shopping experience of choosing an option within a marketplace defined by a finite set of choices [40].

An underlying assumption of estimating customer preferences is that a product can be represented by a group of attributes, and that product attributes can be represented by continuous or discrete variables (i.e. price, color, material, etc.). For the purposes of fielding a CBC survey, all attributes must be discretized into a defined number of attribute levels. Table 2.1 details a set of hypothetical product attributes. The number of attribute levels offered is decided by the designer, who must balance a tradeoff between granularity and survey fatigue.

Table 2.1: Hypothetical Product Attribute Levels

Size	Color	Price
8 in.	Blue	$75
10 in.	Black	$85
12 in.	White	$95
	Silver	

Once a set of product attributes and levels is defined, a CBC survey consisting of choice tasks (i.e. survey questions) is created by creating hypothetical products comprised of a selected level from each attribute. Using set methods for experimental survey design to ensure orthogonality, these tasks are designed to minimize the number of questions that need to be offered in order to capture enough information to fit an accurate model. The task of constructing a survey is often made available by many different software packages, as the development of an orthogonal survey can become increasingly difficult as more attributes and levels are introduced [41,42]. Table 2.2 details a hypothetical choice task composed of the attributes presented in Table 2.1. Note that the respondent has the option to choose

between the three presented hypothetical products or the “None” option. In this example the respondent chose the third product.

Table 2.2: Hypothetical Choice Task

Which product would you choose from these options?			
10 in. Silver $85	12 in. Blue $75	8 in. White $95	None
○	○	⊙	○

After an HB model, or a statistical model that implements Monte Carlo simulations to estimate preference data, is fit to the given survey responses (i.e. preference data), a set of part-worth means are reported. Research in market-based product design has shown that the hierarchical Bayes (HB) mixed logit model captures market heterogeneity more effectively [2,5]. For this reason, an HB model and Sawtooth Software’s CBC/HB module [43] are used to estimate preferences at the respondent level.

A standard representation of these part-worth estimates can be seen in Table 2.3, where the rows correspond to respondents and the columns correspond to attribute levels. It should be noted that part-worth estimates cannot be compared across respondents, but can be compared across attributes for a single respondent to indicate an individual preference towards different attributes. Because a “none” option is presented within both models being used in this research, a part-worth estimate is also provided for this option. For the research presented, part-worth estimates will sum to zero within each attribute (as a means of normalization).

This normalization method is performed within the HB model and is included in most software packages used to estimate part-worths [43].

Table 2.3: Example representation of part-worth estimates for a hypothetical product

		Attributes										None
		Size			Color				Price			
		8 in.	10 in.	12 in.	Blue	Black	White	Silver	$75	$85	$95	
Respondent	1	-1	0	1	3	1	-4	0	4	0	-4	6
	2	-3	1	2	-2	-1	4	-1	3	-2	-1	2
	3	-2	1	1	1	1	-1	-1	3	0	-3	3
	4	-2	0	2	-1	-2	-1	4	5	-2	-3	2

Once a survey is fielded to a group of respondents and selection data is recorded, a random utility model can be estimated under the assumption that respondents are utility maximizers. The estimated part-worths quantify a respondent's preference towards an attribute level and can be combined to determine a respondent's utility for a product. Utility in a compensatory model is described by Equation (2.1), where U_{ij} is respondent i's overall utility for product j. Utility is determined by the sum of an observable utility component V_{ij} and an unobservable utility component ε_{ij}.

$$U_{ij} = V_{ij} + \varepsilon_{ij} \tag{2.1}$$

The observable component of utility V_{ij} is defined by Equation (2.2). It represents the summation of the product of part-worth estimates (β_{ikl}) and active attribute levels (x_{jkl}) for

respondent i, product variant j, attribute k, and level l. For continuous variables (such as price in the given example), interpolation can be used to estimate part-worth estimates between two given levels. The number of product attributes is given by n^a with n_l^k levels per attribute.

$$V_{ij} = \sum_{k=1}^{n^a} \sum_{i=1}^{n_l^k} \beta_{ikl} x_{jkl} \tag{2.2}$$

The unobservable utility component ε_{ij} is an error term that serves to quantify the difference between the model estimation and the respondent's actual decision. Discrete choice models are categorized by the distribution of the unobservable utility component. Normal distributions and extreme value distributions are commonly adopted, leading to formulations of the probit model [44] and logit model [45], respectively. Research has shown that the logit model generally showcases more accurate predictions due to the fat tail (or heavily skewed) assumption of the underlying distribution [38]. Additionally, the logit model is often preferred because of its closed form nature, leading to simplistic share of preference calculations.

Assuming that the error term follows an extreme value distribution, a probabilistic choice rule can be developed [15,17]. Equation (2.3) defines an expression for the probability (p_{ij}) of respondent i choosing product variant j. The probability is calculated by dividing the exponential of product j's utility (V_{ij}) by the sum of the exponentials of the observed utility

for all product variants (n^v) and competing products (n^c), including part-worth estimates for the "none" option, which accounts for the respondent backing out of the market and opting to not purchase any products.

$$p_{ij} = \frac{e^{V_{ij}}}{\sum_{v=1}^{n^v} e^{V_{iv}} + \sum_{c=1}^{n^c} e^{V_{ic}}} \tag{2.3}$$

An alternative to the probabilistic choice rule is the first choice rule. Rather than calculating a probability of purchase based on product utility, the first choice rule models a respondent choosing one product with 100% probability. Stated another way, a respondent is modeled as selecting the product with the highest utility, regardless of the utilities associated with the other products considered in the market simulation. A first choice rule is sometimes preferred as the probabilistic choice rule is sensitive to the introduction of inferior products within the market. This occurs because the Independence of Irrelevant Alternatives (IIA) property is associated with a multinomial logit model. The first choice rule is less sensitive to IIA, rendering it a more robust in market simulations [46]. Both the probabilistic choice and first choice rules are used in this research.

2.2 Market-Based Product Design

Engineering design decisions using customer preference data began with the use of conjoint analysis [14] and the S-model [47]. Initial work in the area of decision-based design primarily focused on estimating designer utility for engineering decisions [44], but later evolved to consider consumer utility to help identify solutions that maximized profit margins

[48]. Choice-based conjoint studies and discrete choice analysis aided design decisions by providing an ability to model consumer heterogeneity using survey instruments formulated around the concept of product selection from a set of alternatives [15–17]. The initial usage of these methods within engineering design led to the first applications of a logit-based model for the design of a single product [49,50].

Product line development research has highlighted the need for more complex models than a standard logit model. This research area was later filled by Li and Azarm [18] who fielded conjoint surveys and fit utility functions at the respondent level, addressing the need for heterogeneity. Recent developments have begun to include continuous representations of heterogeneity within market-based engineering design, such as the nested logit and hierarchical Bayes mixed logit models. The advancements in this area have yielded richer market representations, allowing for more accurate product line optimization opportunities to be explored within the optimization realm.

2.3 Algorithms for Product Line Optimization

Before developing and testing the informed crossover operators, an optimization algorithm needed to be selected. Due to the discrete nature of the product line optimization problem, algorithms that are combinatorial in nature are better suited to assist in the optimization process. Additionally, the product line optimization problem is widely considered to be NP-hard [32], implying that algorithms such as branch-and-bound [51] (that are generally adjusted to solve combinatorial problems) are not well suited for this problem formulation.

Recent research efforts have focused heavily on the use of heuristic algorithms being applied to product line optimization and other similar NP-hard problems. Early efforts focused on the implementation of greedy and other rule-based approaches [21,30,31,33,34,52], detailing that these approaches generated promising results with effective identification of single product solutions. Expanding this problem to a product line requires more robust heuristic methods with more allowable tuning, such as particle swarms [20], simulated annealing [53], and genetic algorithms [54].

This works focuses specifically on the development of operators for a genetic algorithm (GA) [55]. Past research has indicated that a GA is well suited to handle the combinatorial nature of product line design problems [56]. GA's have also demonstrated appealing tradeoffs between objective performance and computational efficiency when compared with other heuristic algorithms [28]. Lastly, a GA can be easily altered to handle problems dealing with different or multiple objectives, making it extremely well suited for this research.

2.4 Genetic Algorithms

A genetic algorithm is a type of heuristic algorithm that consists of five main components: initialization, selection, crossover, mutation, and evaluation. The general process for a GA is outlined in Figure 2.1.

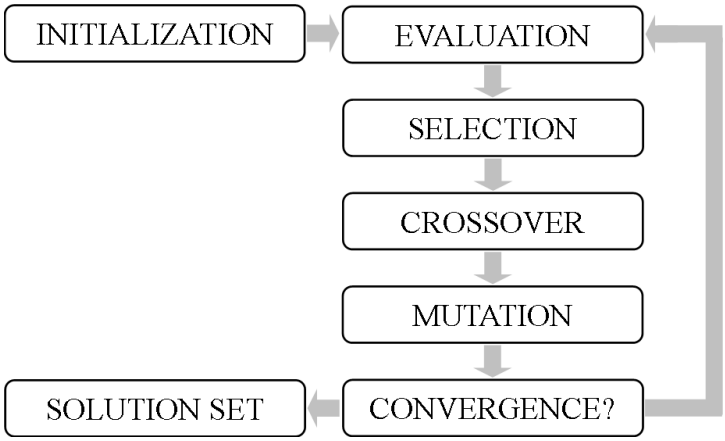

Figure 2.1: Genetic Algorithm Process Diagram

A genetic algorithm (GA) begins by creating a set number of designs to generate an initial population. This population is evaluated against the objective function(s) to determine their initial performance. The selection process then occurs, which selects a pool of designs ("parents") from the population to begin the generation of new designs through reproduction. Reproduction occurs during the crossover process by choosing designs from the selected pool to "cross" via a pre-determined method. These newly generated designs ("children") are then altered slightly through the mutation process, so as to introduce randomness into the overall population. The updated children are returned to the overall population and the performance of the population is re-evaluated and compared against pre-determined convergence criterion. If convergence occurs, the optimal design(s) are chosen, otherwise the population returns to the selection operator and the process continues until convergence.

A standard GA was developed to serve as a baseline for the conducted research. The chosen GA also needed to have the capability to handle multi-objective problems. NSGA-II has been

adopted in the research community as a standard genetic algorithm [57], but new self-adaptive approaches such as the Borg MOEA have also demonstrated promising results when applied to product line optimization problems [58]. For the purposes of this research, the NSGA-II algorithm will serve as a baseline GA due to its prevalence within many programming packages (including MATLAB [59], the programming package used for this research). A random initialization, tournament selection, adaptive mutation, and generational convergence was adopted, and an archiving feature was also encoded that stored past designs to prevent the need for unnecessary re-evaluation.

2.5 Targeted Initial Populations for Market-based Product Line Design Problems

Previous research in the realm of product line optimization and informed genetic algorithm operators consisted of a strategy to improve population initialization [10,60,61]. This approach used customer preference estimates to drive the selection of effective initial design. Referred to as targeted initialization, the first demonstration of this approach identified products that maximized respondent utility for an optimization problem focused on maximizing a product line's share of preference. These optimal products were then combined to create the initial population for a genetic algorithm [10].

This research demonstrated that targeted populations consistently generated product lines that yielded a higher market share of preference when compared to randomly initialized populations [62]. These targeted populations also converged to an optimum at a faster rate, as shown in Figure 2.2.

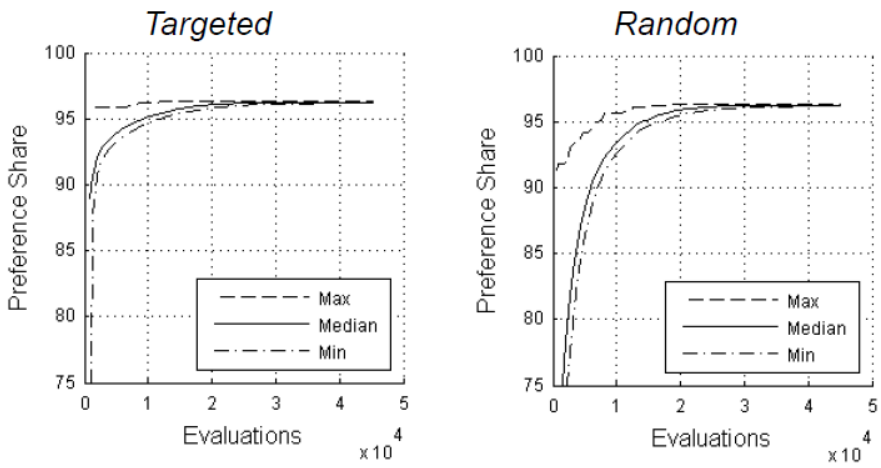

Figure 2.2: Targeted vs. Random Initialization Convergence [63]

The approach described in Figure 2.3 can be extended to both single and multi-objective problem formulations [10]. When considering only a single objective, the targeted objective relates to the main objective (such as maximizing market share of preference). In addition to product configuration variables, price markup variables were included in the problem formulation. These variables indicated the price charged for a feature beyond base cost, and were added to the design string as a floating-point number.

To create the initial population, respondents from the discrete choice survey were selected randomly. For each respondent selected, an objective was targeted (in the case of a two objective problem formulation, this selection was simulated via a coin flip). Using the preference estimates for each respondent, and pre-defined values for the price markup variables, an “ideal” product configuration was created. These individually optimized

products were then combined to seed a product line and product lines were combined to create the initial population [63].

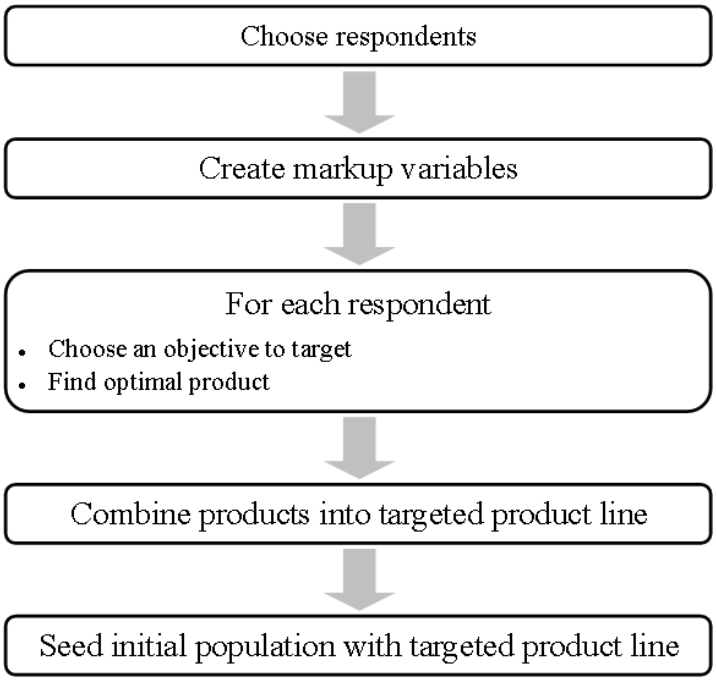

Figure 2.3: Enhanced Targeted Initialization Approach [63]

This informed initialization approach resulted in improved algorithm efficiency and solution quality. When multiple objectives were considered, significant improvements in final hypervolume were achieved when compared to solutions run with an initial populations generated by a Latin hypercube [63].

2.6 Problem-Informed Crossover Operators

Improving the initial population gives the algorithm a better starting point, but design string variations are achieved by the crossover and mutation operators. Previous research efforts

into various other complex optimization problems have altered crossover operators to enhance algorithm performance for a specific problem. Many of these efforts have focused on the traveling salesman problem. Zhou et al. [64] created offspring designs based on comparative parental performance between nodes, while Vahdati et al. [65] compared the distances between two bounding locations of a selected city for both parent designs. Experimental design procedures were used by Ho and Lee [66] to create a level-based technique that employed effects-based data from the parent strings to generate more robust offspring. A real-encoded crossover was proposed by Garcia-Martinez et al. [67] who created offspring within the fitness neighborhood of one parent, while the neighborhood size was defined by the other parent. Others have tailored crossover techniques to suit other specialized optimization problems, such as a capacitated vehicle routing problem [68]. Overall, these modifications improved algorithm effectiveness while preventing premature convergence.

2.7 Chapter Summary

Building on the motivation of these efforts, this research aims to combine the informed operator method established in the targeted population work with a market-based crossover method that uses information from the market domain to improve algorithm performance and solution quality.

CHAPTER 3 – METHODOLOGY

3.1 Introduction

The following chapter will cover an overview of the testing procedure used throughout the remainder of this work. A design string formulation will be established for the optimization algorithm and the calculations for pricing will also be covered. After an overview of these problem basics is completed, the informed crossover operators to be tested will be developed and the genetic algorithm specifics will then be introduced.

3.2 Design String Formulation

The product line optimization problem formulation includes both pricing and feature configuration variables, as both of these influence both market share of preference and overall profit associated with a product line. Product configuration variables represent the different product features included in the product line and take on discrete values to indicate the feature level used in the solution.

For the purposes of this work, a design string representation is defined as depicted in Figure 3.1. A product line will be denoted by a vector consisting of pricing variables (placed at the front of the vector) and feature variables (placed at the back of the vector) [63]. As shown in Figure 3.1, the product line is defined by m pricing variables and n products, where each product has k features. Price variables are encoded as real variables varying between 0 and 1, where a price value of 0 implies that the feature is sold at cost and a 1 indicates a 100%

markup on the corresponding feature. The m price levels are determined by the total number of feature levels modeled in the problem, and are sorted by feature (i.e. if there are two features where feature 1 has two levels and feature 2 has three levels, then there would be 5 pricing variables). Following the m pricing variables, the design string then has $k \times n$ variables to denote the product variant formulations.

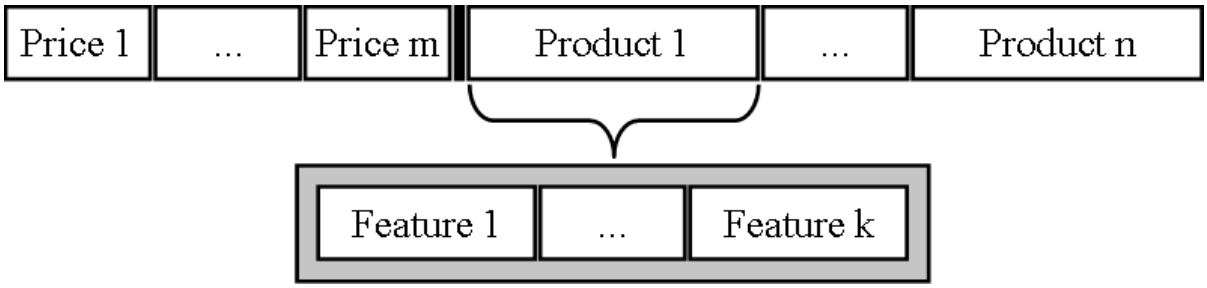

Figure 3.1: Representation of Design String for Product Line Optimization

There are two options for encoding the discretized product configuration variables: as integer variables or as binary representations of the integer variables. To describe the differences between an integer representation of a product's features and a binary representation of a product's features, consider a hypothetical product line containing two product variants as shown in Figure 3.2. Each variant is described by two features, where feature 1 has two levels and feature 2 has three levels. Here, product variant 1 is defined by the second level of feature 1 and the third level of feature 2, while product variant 2 is defined by the first level of feature 1 and the first level of feature 2. The binary representation of this product uses 1's to indicate if a level of a feature is active and 0's if it is inactive. In the example shown in Figure 3.2, the white boxes correspond to feature 1 and the gray boxes correspond to feature

2. The binary representation of the hypothetical product line can be seen beneath the integer representation in Figure 3.2.

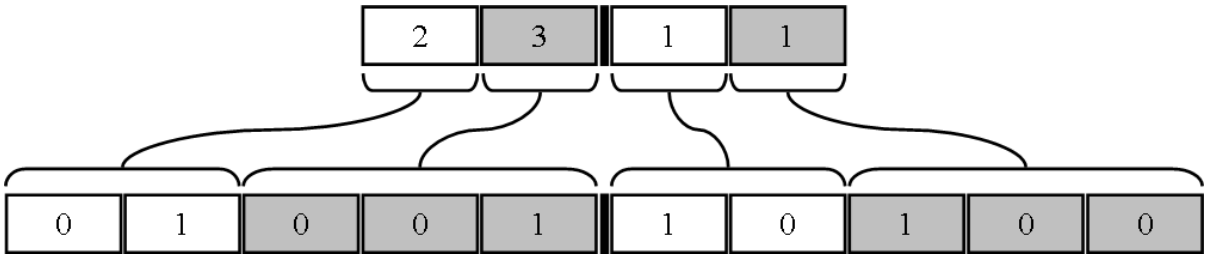

Figure 3.2: Integer and Binary Depictions of a Hypothetical Product Line

3.3 Pricing Calculation

The cost of a product is determined by summing the cost of each feature included in a product with the base price of each product. When calculating the cost of each product from the design string, the product feature string is converted to a binary representation, as shown in Figure 3.2. This binary representation is then multiplied component-wise by the price variable string (as they will be the same length) to yield the overall markup added to the base price. Figure 3.3 provides an example of how corresponding price markups are determined from the design string. In this example, feature 1 has three levels and feature 2 has four levels. The product in question is defined by level 2 of feature 1 and level 3 of feature 2. The arrows in Figure 3.3 denote which price markup value corresponds to the selected feature level.

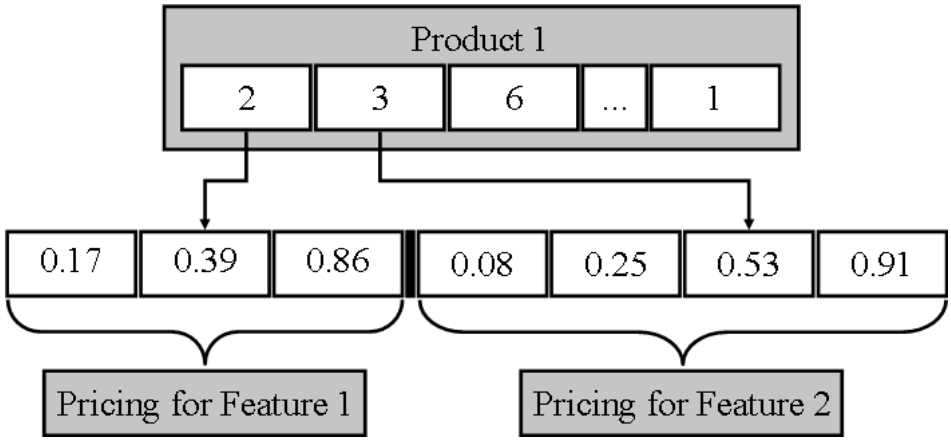

Figure 3.3: Pricing Determination Using Design String

To provide an example of how the pricing calculation is performed, consider Attributes 1 and 2 in Figure 3.3. Equation (3.1) details the component-wise vector multiplication completed to determine the pricing offered by these first two attributes. The resulting vector consists of price markup values to be multiplied by the correlating level costs, and these resulting values are then summed to provide the overall cost of the product.

$$\begin{Bmatrix} 0 \\ 1 \\ 0 \\ 0 \\ 0 \\ 1 \\ 0 \end{Bmatrix} \circ \begin{Bmatrix} 0.17 \\ 0.39 \\ 0.86 \\ 0.08 \\ 0.25 \\ 0.53 \\ 0.91 \end{Bmatrix} = \begin{Bmatrix} 0 \\ 0.39 \\ 0 \\ 0 \\ 0 \\ 0.53 \\ 0 \end{Bmatrix} \tag{3.1}$$

3.4 Objective Development

Three objective functions will be explored: maximizing market share of preference, maximizing profit, and a multi-objective optimization problem that seeks to maximize both objectives simultaneously. Equation (3.2) defines the problem statement for maximizing market share of preference [10].

$$\begin{aligned} &\textbf{Maximize:}\ \text{Share of Preference} \\ &\textbf{Change:}\ \text{Feature levels } (X_{jkl}) \\ &\textbf{Subject To:}\ \text{Price level bounds} \\ &\text{No identical product variants in a line} \\ &\text{Feature level bounds} \end{aligned} \tag{3.2}$$

The share of preference (SoP_j) for a single product variant j can be calculated using Equation (3.3). The share is determined by dividing the total probability of choice (p_{ij}, calculated with Equation (2.3)) for a respondent i by the average value over the entire respondent base (n^r). Share of preference for an entire product line can be found by summing the share of preference for all product variants.

$$SoP_j = \frac{\sum_{i=1}^{n^r} p_{ij}}{n^r} \tag{3.3}$$

Maximizing market share of preference often has adverse effects on profit maximization, as an increase in product utility can be achieved by simply lowering the price of a product. This requires a profit maximization function which simultaneously enforces some degree of

variant desirability. The profitability of a product is approximated using the contribution margin per person in the market. This metric, which is also referred to as per capita contribution margin (PCCM), estimates the profit relating only to a product line, as it does not account for investments, fixed costs, or the time value of money.

The equation for the PCCM metric is defined in Equation (3.4) [10]. The PCCM of a product line is calculated by summing the PCCM of each individual product variant within the line. The PCCM for product j is calculated by finding the margin of the product, or the price of the product (P_j) minus the cost of the product (C_j), and multiplying it by the share of preference for the product (SoP_j) (found with Equation (3.3)).

$$PCCM_j = \left(P_j - C_j\right) * SoP_j \tag{3.4}$$

The problem statement for the profit maximization objective is given by Equation (3.5).

$$\begin{aligned} &\textbf{Maximize:}\ \text{Per Capita Contribution Margin} \\ &\textbf{Change:}\ \text{Feature levels } (X_{jkl}) \\ &\textbf{Subject To:}\ \text{Price level bounds} \\ &\text{No identical product variants in a line} \\ &\text{Feature level bounds} \end{aligned} \tag{3.5}$$

3.5 Crossover Development

Having established a design string formulation, the construction of informed crossover operators can now be covered. Crossover methods were developed so that they included problem specific data, as described in Figure 1.1. Ultimately, each informed crossover

method incorporates market share of preference data to influence their actions (the profit calculation utilizes share of preference data within its calculation). For product line optimization problems, this information includes customer preference data, pricing variables, and attribute level information. Four different crossover methods were created that use the aforementioned problem-specific information:

1. Lowest Share Crossover
2. Lowest k-Share Crossover
3. Mixed Share Crossover
4. Price Sorting Crossover

Each crossover method is thoroughly explained and examined in the following sections.

3.5.1 Lowest Share Crossover

The Lowest Share Crossover Method is driven by a hypothesis that the market share of preference of a product line is significantly lowered by the poorest performing product within the line. By altering this product it is expected that the market share of the entire line will also increase. This operator sorts two product lines in ascending order by relative market share of preference, which is defined as the relative percentage of market share that each product in the line captures with respect to the other products in the line for the purposes of this work. Equation (3.6) describes the relative share of preference of product j in relation to the share of preference of the product line.

$$SoP_{j,relative} = \frac{SoP_j}{SoP} \tag{3.6}$$

The two poorest performing products are then isolated from each product line and scattered crossover is implemented between these two products, as depicted in Figure 3.4.

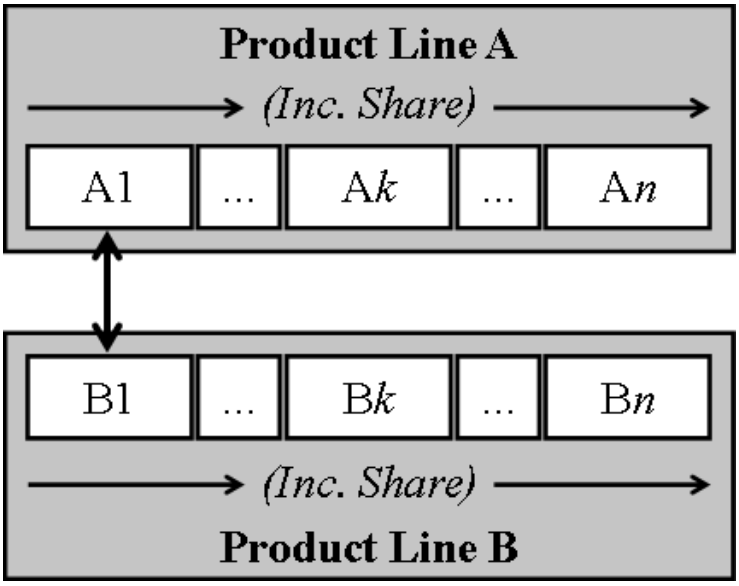

Figure 3.4: Representation of Lowest Share Crossover

A pseudo-code for this method is provided in Figure 3.5.

```
FOR i = 1 : (1/2)*size of population

   Choose rows i and i+1 in selected population

   Calculate market share of preference for both product lines

   Calculate relative market share of preference for each product in each line

   FOR j = 1 : number of products in product line

      Sort product lines in ascending order by relative market share

      Re-index products in both product lines

   END

   Perform scattered crossover on pricing variables

   Perform scattered crossover between first products of each line

END
```

Figure 3.5: Pseudo-Code for Lowest Share Crossover

3.5.2 Lowest *k*-Share Crossover

The Lowest k-Share Crossover method is driven by the hypothesis that overall performance of a product line is driven by its strongest performing products. Similar to the Lowest Share Crossover method, this crossover operator identifies the k poorest performing products (where performance is gauged by the relative market share of preference) to be altered via crossover. This crossover method again sorts two product lines in ascending order by their relative market share of preference. The lowest k products (where k is an integer ranging

from 2 to n for a product line with n products) are then selected and scattered crossover occurs between corresponding products.

A visual depiction of the Lowest k-Share Crossover Method can be seen in Figure 3.6.

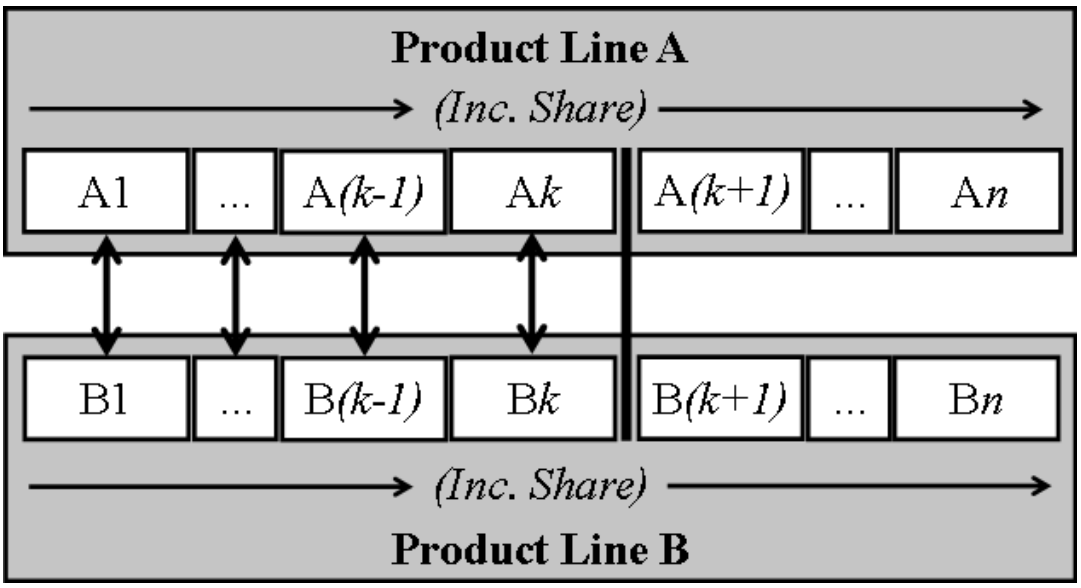

Figure 3.6: Respresentation of Lowest k-Share Crossover

A pseudo-code for this method is provided in Figure 3.7.

```
FOR i = 1 : (1/2)*size of population

   Choose rows i and i+1 in selected population

   Calculate market share of preference for both product lines

   Calculate relative market share of preference for each product in each line

   FOR j = 1 : number of products in product line

      Sort product lines in ascending order by relative market share

      Re-index products in both product lines

   END

   Perform scattered crossover on pricing variables

   FOR k = 1 : selected number of products

      Perform scattered crossover between products 1 through k from each line

   END

END
```

Figure 3.7: Pseudo-Code for Lowest k-Share Crossover

3.5.3 Mixed Share Crossover

The Mixed Share Crossover method is motivated by the hypothesis that mixing the best and worst performing products in a product line will lead to a stronger design string. By “mixing” two product lines, there is a possibility of equilibrating the two product lines so that further niches can be explored. The Mixed Share Crossover scheme takes two design strings, sorting one product line in ascending order and one product line in descending order (by relative

market share of preference). Scattered crossover then occurs between each corresponding product, as shown in Figure 3.8.

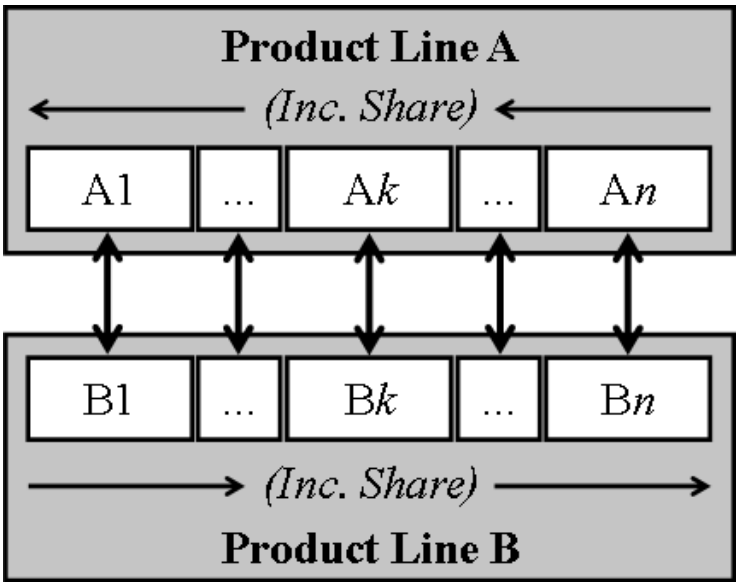

Figure 3.8: Representation of Mixed Share Crossover

A pseudo-code for this method is provided in Figure 3.9.

```
FOR i = 1 : (1/2)*size of population

   Choose rows i and i+1 in selected population

   Calculate market share of preference for both product lines

   Calculate relative market share of preference for each product in each line

   FOR j = 1 : number of products in product line

      Sort product lines in ascending order by relative market share

      Re-index products in both product lines

   END

   Perform scattered crossover on pricing variables

   FOR k = 1 : number of products in line

      Perform scattered crossover between products 1 through k from each line

   END

END
```

Figure 3.9: Pseudo-Code for Mixed Share Crossover

3.5.4 Price Sorting Crossover

The Price Sorting Crossover method uses product price calculated from the pricing variables and product configuration profiles to sort a design string. This clusters similarly priced items, with the motivating concept being that items in similar price categories may have shareable attributes that can be swapped to maximize line performance. Scattered crossover then occurs between products in corresponding similarly priced products, as depicted in Figure 3.10.

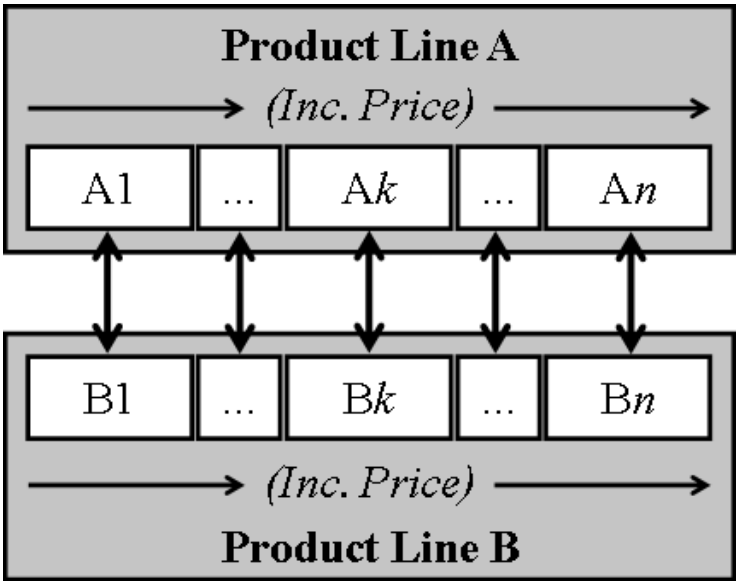

Figure 3.10: Representation of Price Sorting Crossover

A pseudo-code for this method is provided in Figure 3.11.

```
FOR i = 1 : (1/2)*size of population

   Choose rows i and i+1 in selected population

   Calculate price of each product in each product line

   FOR j = 1 : number of products in product line

      Sort product lines in ascending order by price

      Re-index products in both product lines

   END

   Perform scattered crossover on pricing variables

   FOR k = 1 : selected number of products

      Perform scattered crossover between products 1 through k from each line

   END

END
```

Figure 3.11: Pseudo-Code for Price Sorting Crossover

3.6 Testing Procedure

For the purposes of testing, certain aspects of the genetic algorithm were maintained as constant throughout all trials. The constant parameters of the algorithm are listed below.

- Population Size: fixed at two times the number of design variables
- Selection: tournament with four candidates
- Crossover Rate: fixed at 0.8
- Mutation: uniform at a rate of 0.05
- Evaluation: only retain unique product lines [69]
- Convergence: 600 generations

The problem, objective function, size of the product line, initialization method, and crossover method were all altered each experiment, and 5 – 10 trials were run per experiment. A collection of experimental designs consisting of combinations of the aforementioned parameters was constructed using the different options presented in Table 3.1. The parameters listed in this table were chosen to test the validity of the developed informed crossover operators. The following chapters are divided based on components of this experimental design: Chapter 4 covers the single objective MP3 Problem, Chapter 5 the single objective Vehicle Problem, and Chapter 6 is an overview of the results from a multi-objective optimization of the Vehicle problem. In total, 800 trials are conducted on the MP3 problem, 1,600 on the single objective Vehicle problem, and 160 trials were conducted on the multi-objective Vehicle problem (for a total of 2,560 trials).

Table 3.1: Experimental Design Parameters

Problem	Objective	Number of Products	Initialization	Market Model	Crossover
MP3	Market Share	4 Products	Random	Probabilistic Choice	Lowest Share
Vehicle	Profit	5 Products	Targeted	First Choice	Lowest k-Share
	Multi-Objective	6 Products			Mixed Share
		7 Products			Price Sorting
					Scattered

To determine the validity of the test trials, a baseline was established for each combination of problem, objective function, and product line size. The baseline experimental design uses a random initialization and scattered crossover. For single objective trials, the average and standard deviation of both the objective performance and the generations required for the strongest performing design to stagnate (referred to as stall generations) are compared to the baseline model. These two measures served to determine effectiveness and efficiency, respectively. The measures used to analyze the multi-objective trials are introduced in Section 6 as they relate to the various newly created values introduced with multi-objective optimization.

The following chapters will discuss the results from these test trials, beginning with the objective and generational results from the single objective MP3 problem.

CHAPTER 4 – MP3 PROBLEM

4.1 Problem Introduction

The first case study analyzed in this research concerns the optimization of a product line consisting of MP3 players. The preference model for the MP3 player problem was constructed from a choice-based conjoint survey fielded to 205 respondents. Choice task questions were based on the 12 attributes detailed in Table 4.1. The price for each feature level is noted in Table 4.2. Sawtooth's CBC/HB software [43] was used to estimate the part-worth coefficients for each respondent. The "none" option was also included in the choice tasks, and its part-worth was also estimated.

Marketplace competition was included in the MP3 market simulation to increase the difficulty of locating an optimal product line and to more accurately simulate a "real-world" market scenario. These competitive products were configured by running an optimization for a five product line solution against the outside good. The configurations of these products are shown in Table 4.3.

It should be noted that due to the relatively low number of product combinations (393,216 total combinations) a valid solution can be found relatively easily with standard genetic algorithm operators. This allows the MP3 product line development problem to be used to determine if (a) the developed crossover operators negatively affect the optimization, and (b) if the informed crossover operators render any improvements in computational efficiency.

Table 4.1: MP3 Player Attributes and Price Levels [10]

Level	Photo/Video/ Camera	Web/App/ Ped	Input	Screen Size	Storage	Background Color	Background Overlay	Base Price
1	None	None	Dial	1.5 in diagonal	2 GB	Black	No Pattern/Graphic Overlay	$49
2	Photo Only	Web Only	Touchpad	2.5 in diagonal	16 GB	White	Custom Pattern Overlay	$99
3	Video Only	App Only	Touchscreen	3.5 in diagonal	32 GB	Silver	Custom Graphic Overlay	$199
4	Photo and Video Only	Ped Only	Buttons	4.5 in diagonal	64 GB	Red	Custom Pattern and Graphic Overlay	$299
5	Photo and Lo-Res Camera	Web and App Only		5.5 in diagonal	160 GB	Orange		$399
6	Photo and Hi-Res Camera	App and Ped Only		6.5 in diagonal	240 GB	Green		$499
7	Photo, Video, and Lo-Res Camera	Web and Ped Only			500 GB	Blue		$599
8	Photo, Video, and Hi-Res Camera	Web, App, and Ped			750 GB	Custom		$699

Table 4.2: MP3 Player Cost per Feature [10]

Level	Photo/Video/Camera	Web/App/Ped	Input	Screen Size	Storage	Background Color	Background Overlay
1	$0.00	$0.00	$0.00	$0.00	$0.00	$0.00	$0.00
2	$2.50	$10.00	$2.50	$12.50	$22.50	$5.00	$2.50
3	$5.00	$10.00	$20.00	$22.50	$60.00	$5.00	$5.00
4	$7.50	$5.00	$10.00	$30.00	$100.00	$5.00	$7.50
5	$8.50	$20.00		$35.00	$125.00	$5.00	
6	$15.00	$15.00		$40.00	$150.00	$5.00	
7	$16.00	$15.00			$175.00	$5.00	
8	$21.00	$25.00			$200.00	$10.00	

Table 4.3: MP3 Player Competition Design [10]

	Product 1	**Product 2**	**Product 3**	**Product 4**	**Product 5**	**None**
Photo/Video/ Camera	Photo, Video, and Hi-Res Camera	Photo, Video, and Hi-Res Camera	Photo, Video, and Hi-Res Camera	Photo, Video, and Hi-Res Camera	Photo, Video, and Hi-Res Camera	N/A
Web/App/ Ped	Web and App Only	Web and App Only	Web and App Only	Web, App, and Ped	Web, App, and Ped	
Input	Dial	Touchscreen	Touchscreen	Touchscreen	Touchscreen	
Screen Size	1.5 in Diagonal	4.5 in Diagonal	4.5 in Diagonal	4.5 in Diagonal	6.5 in Diagonal	
Storage	16 GB	16 GB	16 GB	64 GB	160 GB	
Background Color	Silver	Silver	Silver	Custom	Green	
Background Overlay	Custom Pattern and Graphic Overlay	Custom Graphic Overlay	Custom Pattern and Graphic Overlay	Custom Pattern and Graphic Overlay	Custom Graphic Overlay	
Price	*$132.59*	*$211.39*	*$216.39*	*$438.89*	*$504.14*	*$0.00*
Preference Share	*25%*	*27%*	*20%*	*15%*	*10%*	*3%*

4.2 Tuning the Lowest k-Share Crossover

Before running the experimental design (consisting of a full factorial combination of objective function, product line size, market model, initialization method, and crossover operator), testing was completed to determine an optimal relationship for the value k associated with the Lowest k-Share Crossover method. Ten trials were conducted using the baseline GA settings with a market share maximization objective, a probabilistic choice rule, random initialization, the Lowest k-Share Crossover method, and product line sizes of 5, 6, and 7 products (a product line size of 4 products was removed due to the overly simplistic nature of the problem). The market share objective was chosen due to the crossover method's usage of market data. For each product line containing n products (for $n \in [5,6,7]$), k was varied from 1 to n. The average market share was determined over the ten trials for each value of k, and the results of these experiments are presented in Figure 4.1.

From Figure 4.1 it can be seen that improved algorithm performance was often associated with k values equal to $n - 1$. It can be theorized that the improved performance of this crossover method was due to the fact that the strongest product in each line is left unaltered, and the remaining products are subsequently changed through scattered crossover. The conceptual basis of this is similar to the reasoning behind the Lowest Share Crossover method, which aims to only alter the poorest performing product based on market share. Given these findings, $k = n - 1$, for the remainder of this thesis, and Lowest k Share Crossover will be referred to as the Lowest $n - 1$ Share Crossover method.

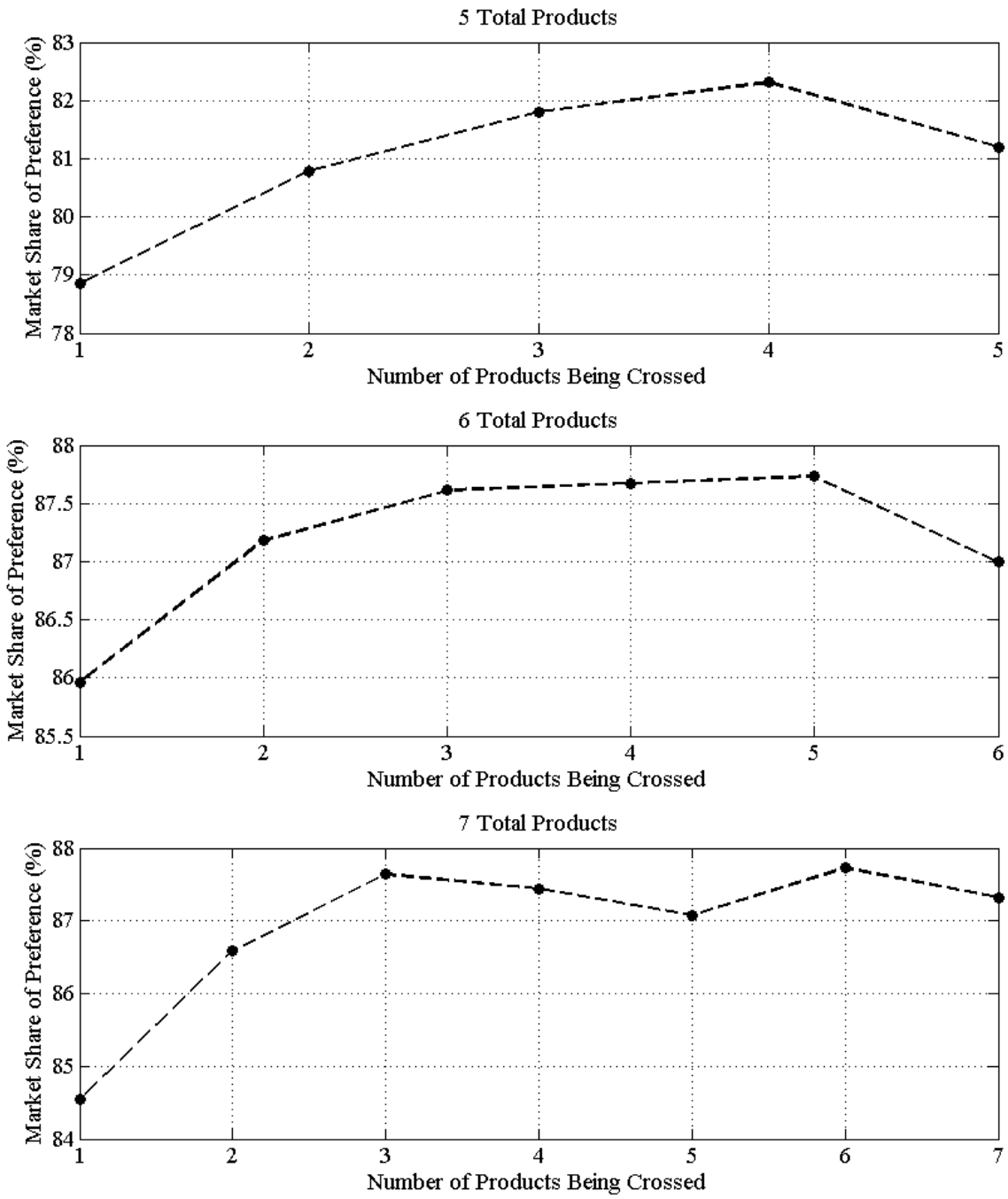

Figure 4.1: Results of Testing Varying *k* Values across Different Product Line Sizes

4.3 Full Factorial Experimental Design Trial Results

The four crossover methods previously described were encoded in MATLAB [59], including the baseline scattered crossover. These operators were incorporated into a genetic algorithm with the experimental standards presented below for population size, selection operator, crossover rate, mutation operator, mutation rate, evaluation method, and convergence criteria.

- Population Size: fixed at two times the number of design variables
- Selection: tournament with four candidates
- Crossover Rate: fixed at 0.8
- Mutation: uniform at a rate of 0.05
- Evaluation: only retain unique product lines [69]
- Convergence: 600 generations

To account for the heuristic nature of a genetic algorithm, ten trials were conducted for each combination of the full factorial design from Table 3.1 (consisting of every combination of objective function, market model, and crossover method). Objective function values and stall generations were collected for each trial. Random initialization was only used with the MP3 product line problem to remove any positive effects contributed by a targeted population and to allow for a full investigation of the effects of the informed crossover methods. Multi-objective trials were not conducted for this problem.

The following sections are subdivided by the objective used to maximize the problem in order to analyze the results collected from the test trials. Plots of all relevant data are also presented in these subsections.

4.3.1 Maximizing Market Share of Preference

Data was collected for the ten trials being run with the market share objective concerning objective and generational performance. The results of these trials are presented in Figure 4.2 – Figure 4.5. Figure 4.2 presents the objective results using the probabilistic choice rule, Figure 4.3 presents the generational results using the probabilistic choice rule, Figure 4.4 presents the objective results using the first choice rule, and Figure 4.5 presents the generational results using the first choice rule.

The following figures present the relevant data using box plots. A box plot is composed of two components: a box and whiskers. The box presents the median value (the middle red line) and the upper and lower quartiles of the data (bounding the edges of the box). The upper lower datums are presented as the whiskers, or tail ends of the boxes. If a data point falls outside of 1.5 times the inner-quartile range (or the upper quartile minus the lower quartile), it is presented as an outlier and is not considered in the maximum and minimum values for the whiskers.

Figure 4.2 indicates that the developed informed crossover operators do not have a negative impact on the objective performance of the space. Every crossover method reaches optimal performance, as indicated by the asymptotic behavior of the box plots presented within these figures. This behavior was expected, as a traditional GA was capable of locating the optimal solutions for these product line sizes when market share was being maximized. The variations in the plots are due to the probabilistic nature of heuristic algorithm solutions, and it is theorized that with ample time and resources, enough sampling could be conducted to construct a reasonable spread of data around a definable mean.

Having defended the ability of the informed crossover operators to locate an optimal point using the probabilistic first choice rule, the computational efficiency of these operators can now be analyzed. Figure 4.3 presents the average number of generations needed to reach convergence for each of these crossover methods (where methods with higher computational efficiency will minimize the generations required to reach convergence). Analyzing this figure reveals that as product line sizes increase, the computational efficiency of all the developed crossover methods increases when compared to scattered crossover. The same experimental setup will be applied using the first choice rule to determine the efficacy of these informed crossover operators when applied to a different choice model.

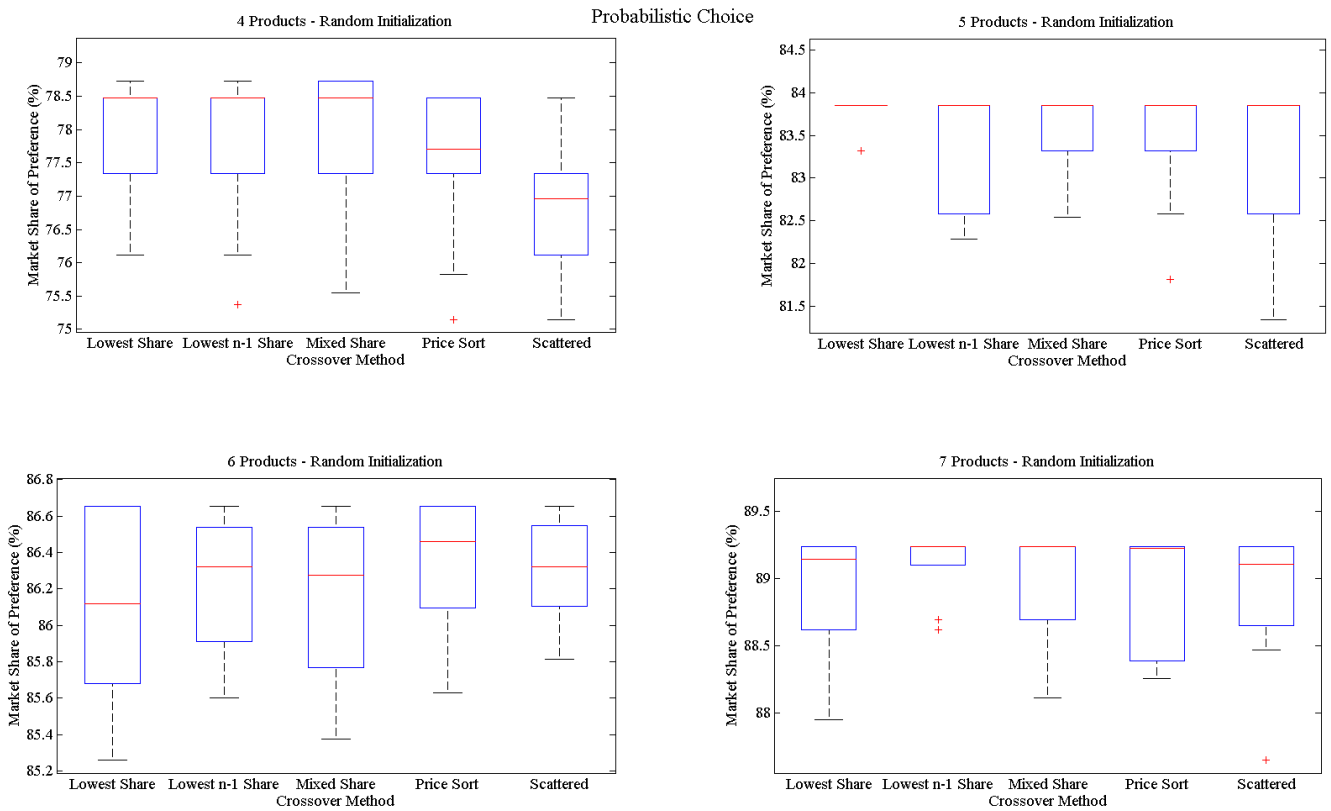

Figure 4.2: Objective Function Results for MP3 Case Study Using a Probabilistic Choice Rule and Maximizing Market Share of Preference (Larger is Better)

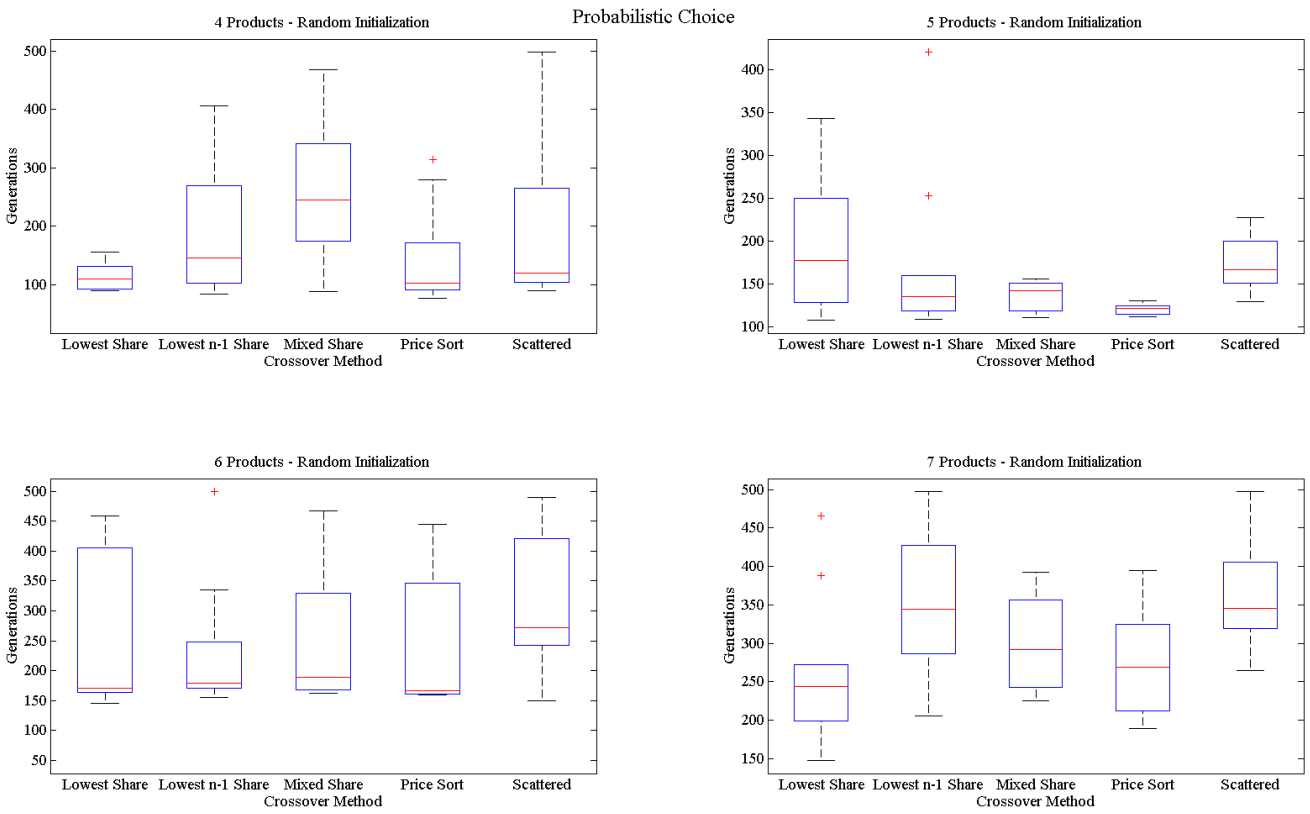

Figure 4.3: Generational Results for MP3 Case Study Using a Probabilistic Choice Rule and Maximizing Market Share of Preference (Smaller is Better)

Figure 4.4 provides the objective results from the trials conducted on the MP3 problem using a first choice rule. Similar to the trials conducted with the probabilistic choice rule, each crossover method reaches optimal performance. This indicates the informed crossover operator abilities to arrive at an optimum under a more robust choice rule. Similar variations in objective performance are observed, but this is again due to the inherently random nature of a GA.

Figure 4.5 present the average number of generations needed to reach convergence for each of these crossover methods (where methods with higher computational efficiency will minimize the generations required to reach convergence). Disregarding slight performance discrepancies with product lines consisting of 4 products (when using the Mixed Share crossover method), all informed crossover operators provide improvements in the number of generations required to converge on an optimum. This again suggests that the informed crossover methods are robust enough to handle larger problems and more robust choice models.

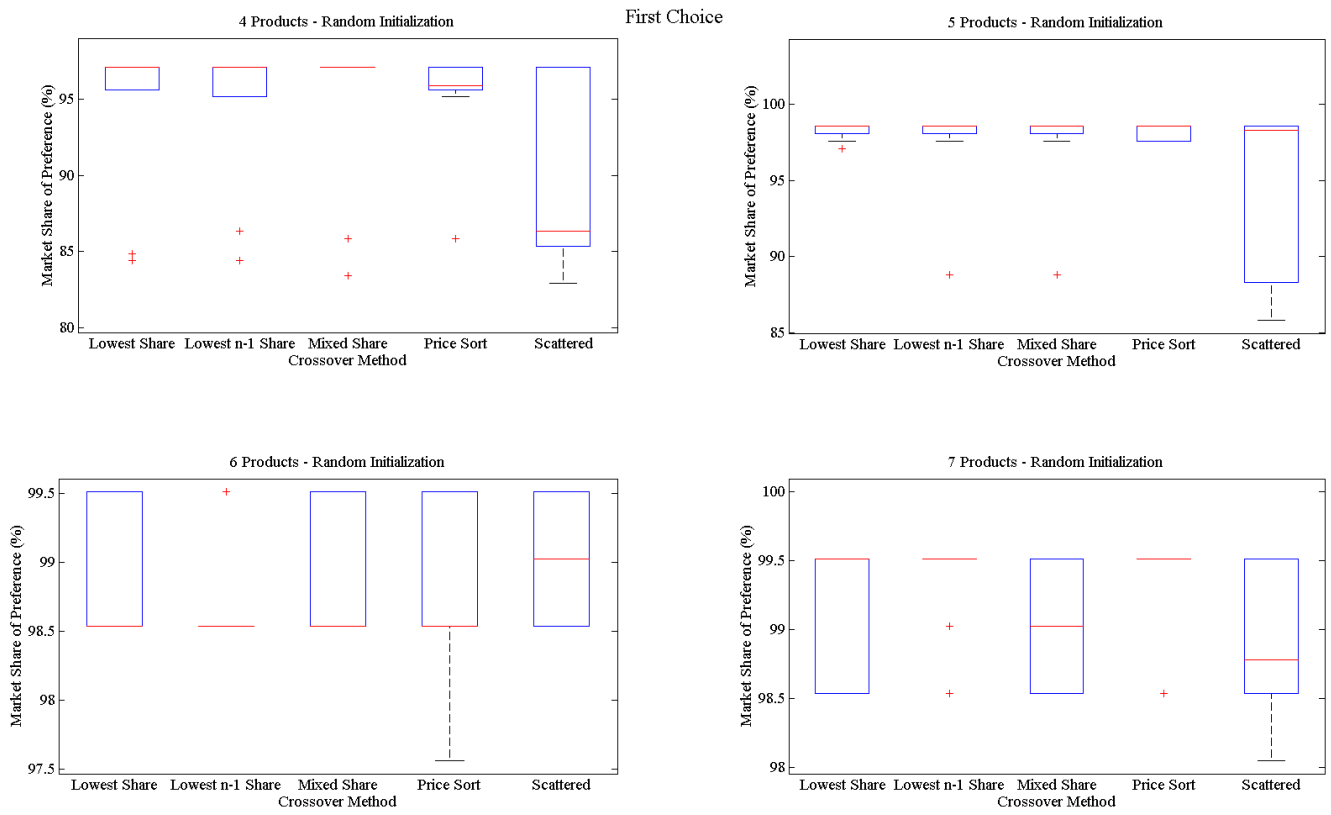

Figure 4.4: Objective Results for MP3 Case Study Using a First Choice Rule and Maximizing Market Share of Preference (Larger is Better)

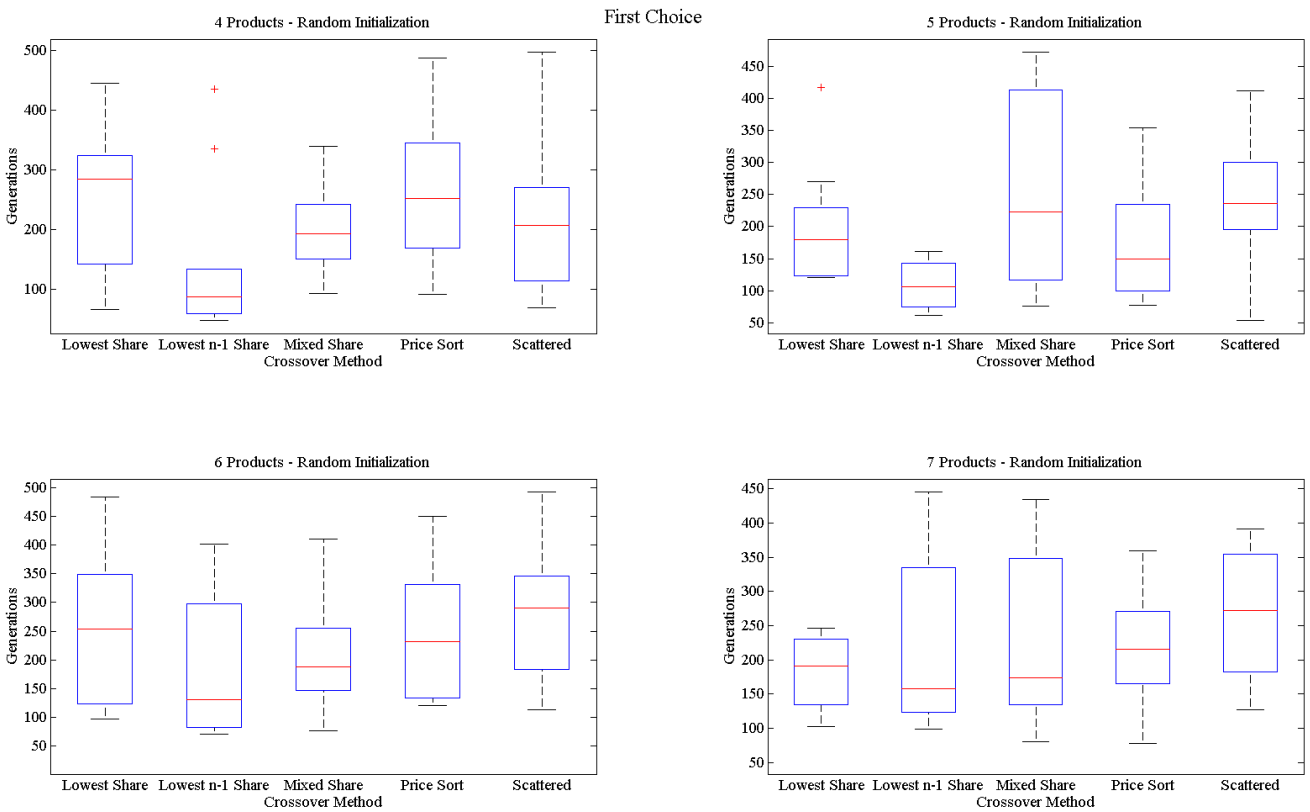

Figure 4.5: Generational Results for MP3 Case Study Using a First Choice Rule and Maximizing Market Share of Preference (Smaller is Better)

An overview analysis of the findings from all trials conducted with the MP3 problem maximizing market share of preference indicates strong performance by the informed crossover operators. They offer improvements in algorithm efficiency without sacrificing solution quality. This suggests that the informed crossover methods are robust enough to handle larger problems (a product line composed of 7 products would have 2.88E+35 total possibilities). To help support these findings, the same experimental setup will be applied to the profit maximization problem.

4.3.2 Maximizing PCCM

Data was collected for the ten trials being run with the PCCM objective concerning objective and generational performance. The results of these trials are presented in Figure 4.6 – Figure 4.9. Figure 4.6 presents the objective results using the probabilistic choice rule, Figure 4.7 presents the generational results using the probabilistic choice rule, Figure 4.8 presents the objective results using the first choice rule, and Figure 4.9 presents the generational results using the first choice rule.

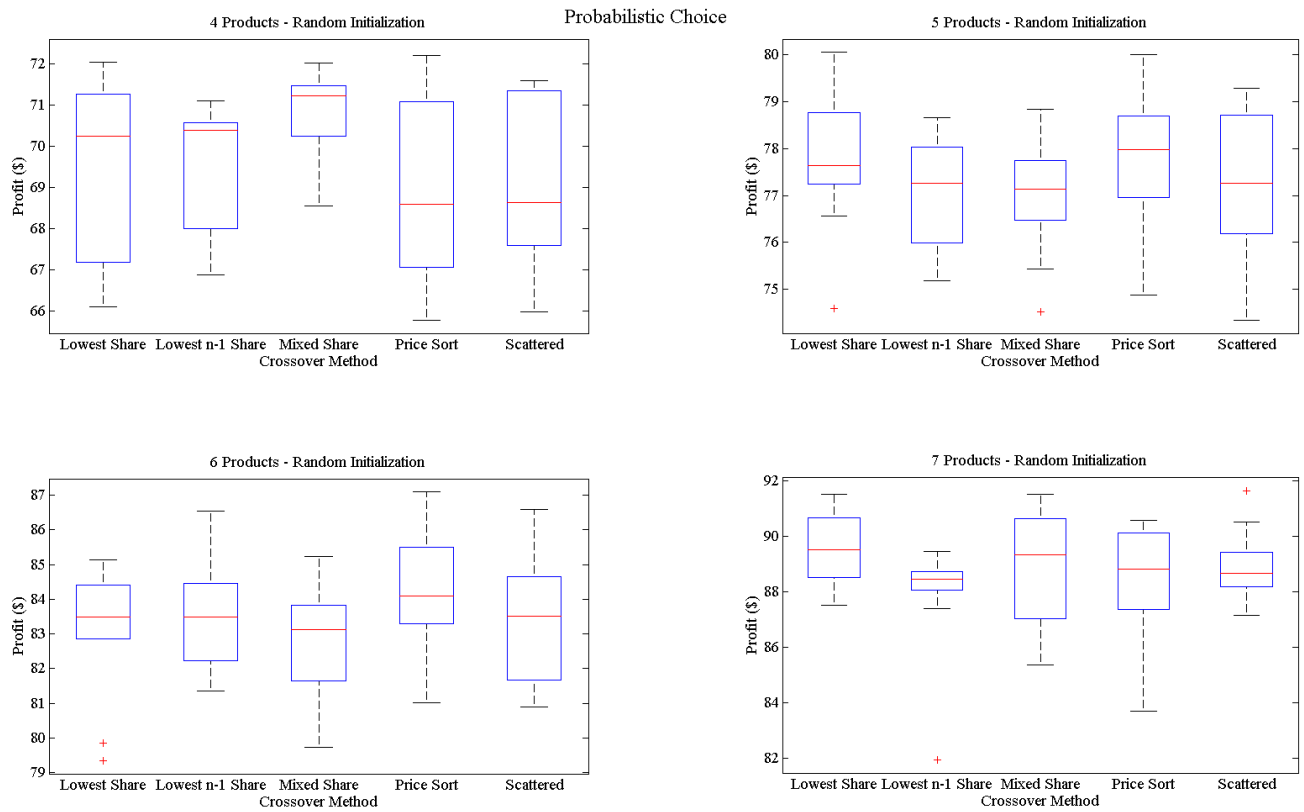

Figure 4.6: Objective Results for MP3 Case Study Using a Probabilistic Choice Rule and Maximizing PCCM (Larger is Better)

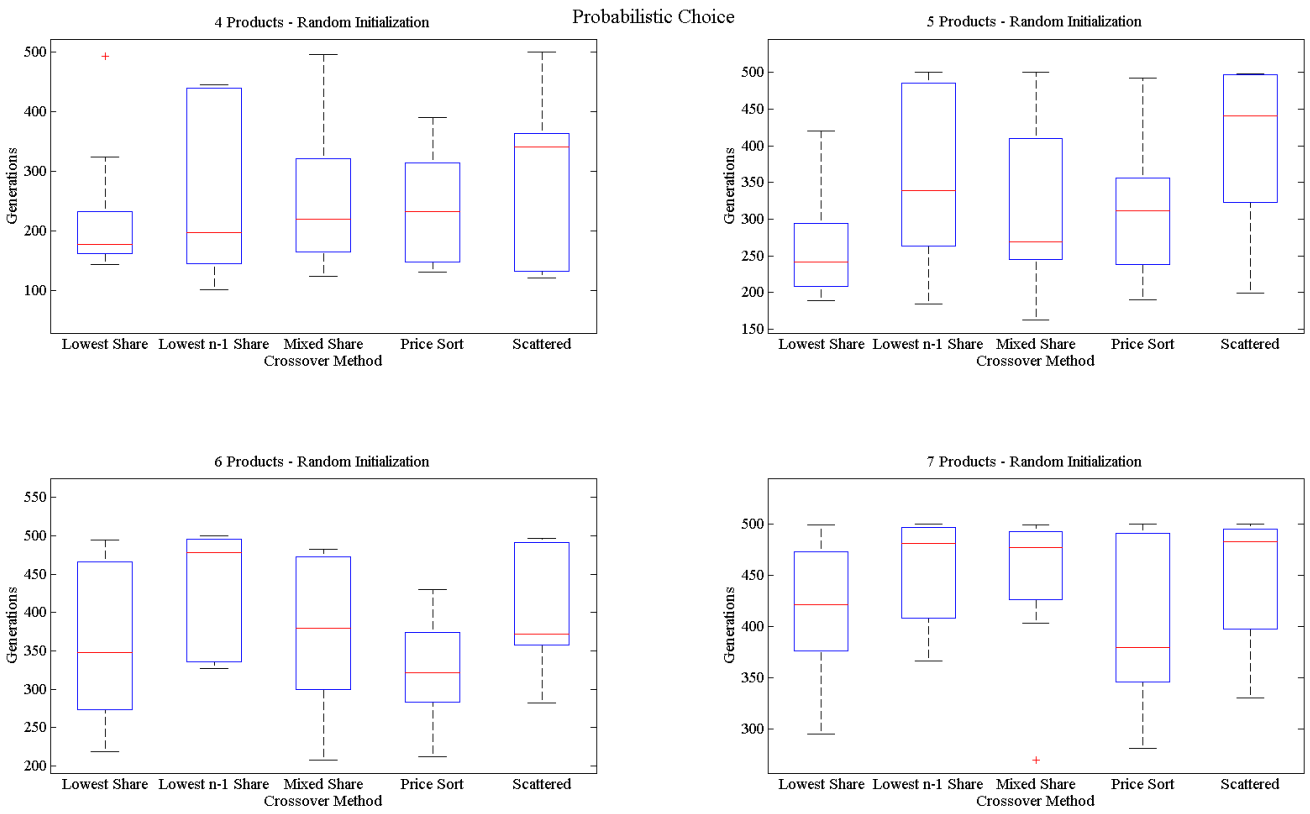

Figure 4.7: Generational Results for MP3 Case Study Using a Probabilistic Choice Rule and Maximizing PCCM (Smaller is Better)

Due to the more complex nature of the PCCM maximization problem (as it combines both share of preference data and pricing data, the algorithm does not demonstrate as clear results as the share of preference objective following analysis of the data. When examining the results from the use of the probabilistic choice rule, it can be seen in Figure 4.6 that objective improvement is provided by each of the informed crossover operators. Consistent objective performance is provided by the Price Sort Crossover, as would be expected when maximizing the profit of the product line (due to the incorporation of product price within the Price Sort Crossover method).

Analysis of the generational performance provided by the informed crossover methods when used with the probabilistic choice rule (in Figure 4.7) again demonstrates near consistent improvements in computational efficiency. The one contradiction to this statement occurs with the Lowest $n - 1$ Share Crossover for a product line with 6 products. This spread is due to the inherent randomness associated with a GA and further test trials conducted with this step would reverse this behavior (yielding overall improvements in computational efficiency).

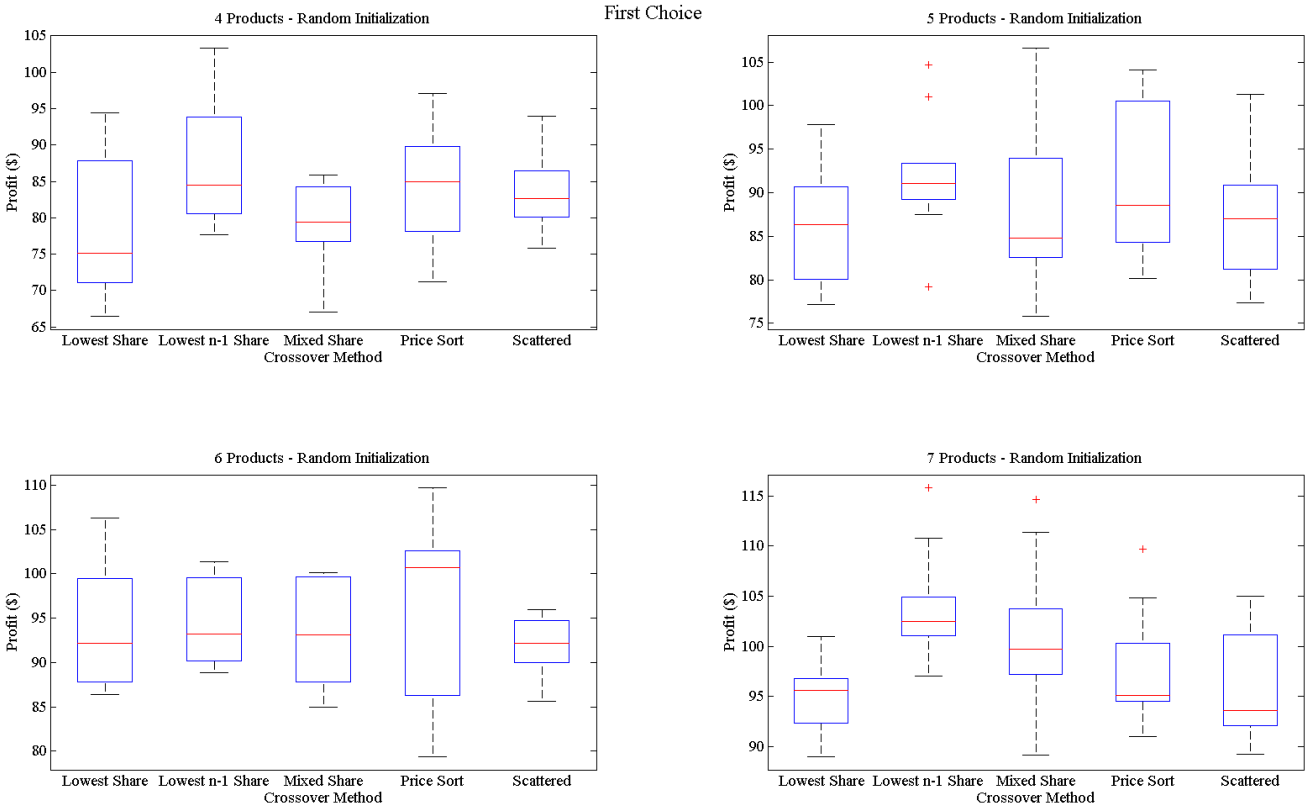

Figure 4.8: Objective Results for MP3 Case Study Using a First Choice Rule and Maximizing PCCM (Larger is Better)

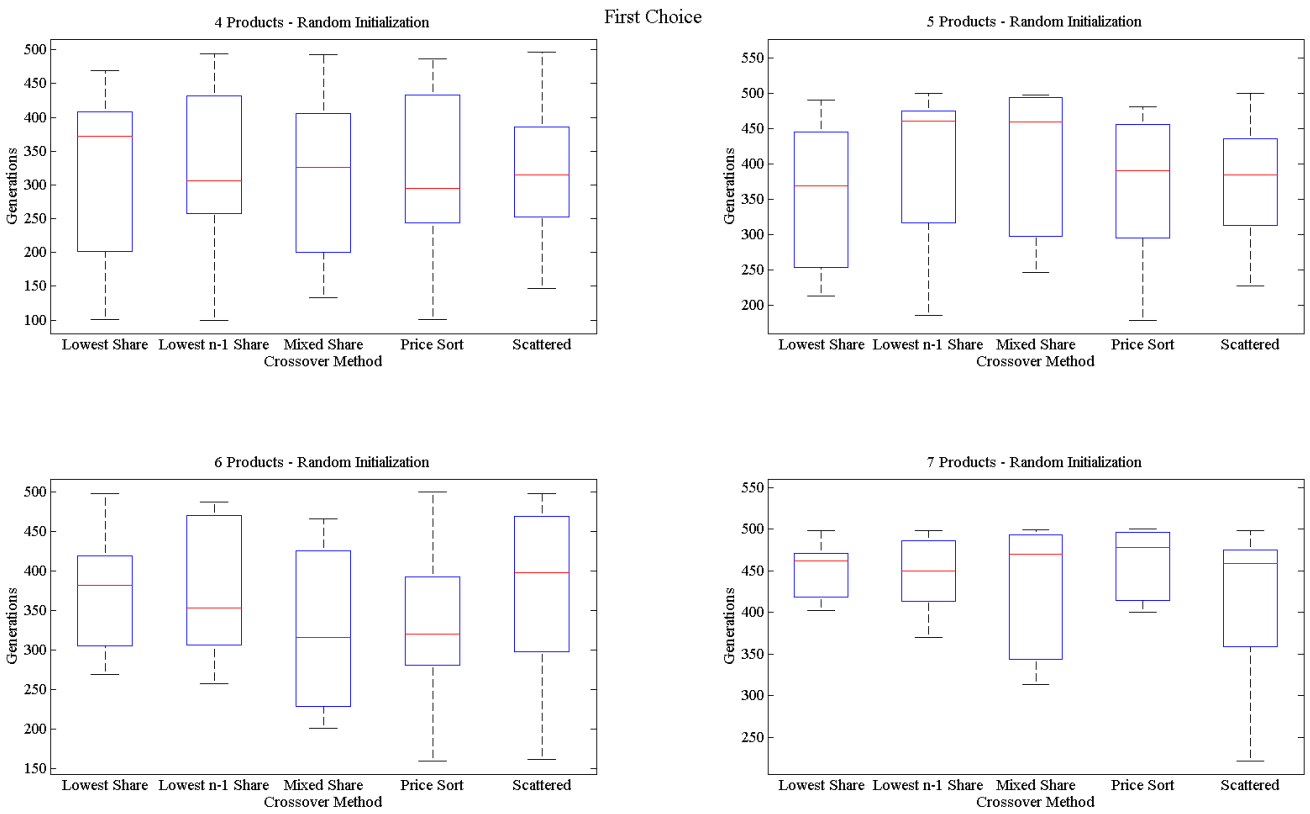

Figure 4.9: Generational Results for MP3 Case Study Using a First Choice Rule and Maximizing PCCM (Smaller is Better)

Analysis of the informed crossover methods when applied with the first choice rule reveal slightly different results. Algorithm effectiveness is not demonstrated by the informed crossover methods until product line sizes increase past five products. This could be due to the fact that the first choice rule creates a design space that is more robust, or more difficult to locate an optimum in, due to the binary nature of the rule. Small changes to a product will have less of an impact on the overall share of the product line when compared with share calculated using the probabilistic choice rule, due in part first choice rule's insensitivity to IIA. By having product lines with less than 5 products, there are less options for respondents to choose from, ultimately making it more difficult to create products that respondents will want to migrate towards.

Despite these issues, it can be theorized that these crossover methods are more robust than the standard scattered crossover, as they show improvements with a more rigorous market model and larger scale problems (as evidenced by the data presented in Figure 4.8). Analysis of Figure 4.9 demonstrates the computational gains provided by the informed crossover operators. In contrast to the probabilistic choice rule, it seems that the developed crossover operators simply match computational performance when compared with the baseline scattered crossover. No significant generational spreads are noted in Figure 4.9, indicating that the informed crossover operators are comparable to scattered crossover.

4.4 Chapter Summary

Overall analysis of the initial test trials conducted with the MP3 problem indicates either slight computational improvement or generational improvement provided by the informed crossover methods. Considering the fact that the MP3 product line design optimization problem is relatively simplistic and can be solved with a standard GA, these results prove that the developed crossover operators provide some indication of improvement when applied to product line optimization problems and should be tested on more rigorous problems to determine the benefits they can provide. Further test trials will explore the effectiveness of these informed crossover operators when applied to a more complex product line optimization problem (that standard genetic algorithms often struggle with).

CHAPTER 5 – VEHICLE PROBLEM

5.1 Problem Introduction

In order to test the full range of benefits offered by the informed crossover operator, a more complex product line optimization problem is needed. The automobile packaging problem concerns a product line constructed of various automobiles and is based on customer preference data that was developed from a choice-based conjoint survey fielded to 2,275 respondents [10]. The observed part-worth estimates were established using an HB mixed logit model that was fit using the Sawtooth CBC/HB [43] module. Each respondent in the HB model had 73 total part-worth estimates: 60 for the features offered, 12 for the price levels, and an estimate for the "none" option (i.e. the outside good).

Due to the proprietary nature of the problem, the feature and cost breakdowns cannot be fully included in this document. They are, however, similar in nature to the breakdowns shown in Table 4.1 and Table 4.2. The total number of levels available for each of the 19 attributes pertaining to the automobile packaging problem, as well as the listed price markups, are shown in Table 5.1. Note that the price markup values are normalized according to the maximum possible price level and are presented in the form $(Markup\ 1\ |\ldots|\ Markup\ n\ )$. For example, Attribute 1 has 3 levels, and the price markups corresponding to each of those three variables are 0.8, 2.2, and 3.2. The base cost will be assumed to be a normalized $0.00. Accounting for all possible feature combinations yields 1,074,954,240 total product possibilities, justifying this product line development problem as both large-scale and

complex. Due to the complexity of this problem, the outside good will serve as the source of competition when optimizing product lines in the market simulator.

Table 5.1: Automobile Feature Levels per Attribute [10]

Attribute Number	Number of Levels	Price Markup
1	3	(0.8 | 2.2 | 3.2)
2	2	(0 | 0)
3	5	(5 | 5 | 5 | 5.1 | 0)
4	6	(0.5 | 1.3 | 1.4 | 1.9 | 2.7 | 0)
5	2	(0.5 | 0)
6	3	(0.6 | 3.3 | 0)
7	3	(1.1 | 2.4 | 0)
8	2	(2.2 | 0)
9	4	(2.3 | 3 | 3.1 | 0)
10	2	(1.9 | 0)
11	3	(2.2 | 5 | 0)
12	2	(0.5 | 0)
13	4	(1 | 1 | 2.1 | 0)
14	3	(0.1 | 0.6 | 0)
15	3	(5.1 | 5.3 | 0)
16	4	(0.4 | 0.6 | 0.6 | 0)
17	4	(1.3 | 1.3 | 2.7 | 0)
18	3	(0.9 | 5.9 | 0)
19	2	(2 | 0)

The four crossover methods previously described were encoded in MATLAB, as well as the base scattered crossover. These operators were incorporated into a genetic algorithm with the listed experimental standards for population size, selection, crossover rate, mutation, mutation rate, evaluation method, and convergence criteria.

- Population Size: fixed at two times the number of design variables
- Selection: tournament with four candidates
- Crossover Rate: fixed at 0.8
- Mutation: uniform at a rate of 0.05
- Evaluation: only retain unique product lines [69]
- Convergence: 600 generations

To account for the heuristic nature of a genetic algorithm, ten trials were run for each combination of the full factorial design and the objective values and stall generations were collected for each trial. The full factorial test trial collection resulted in 1,600 total trials, each conducted for 600 generations.

The following sections are subdivided by objective function to allow for a structured analysis of the data collected from the conducted test trials. The full set of figures created from the aforementioned test trials can be found in Appendix A.

5.2 Maximization of Market Share of Preference of the Vehicle Problem

The maximization of market share of preference within the vehicle product line design problem yielded numerous trends indicating the benefits of informed crossover operators. Due to the large amount of data collected and presented within these trials, only product lines containing 6 and 7 products are presented in this chapter. The remaining results are provided in Appendix A. specific product line sizes corresponding to combinations of market model and initialization method are presented that highlight the discussed trends.

5.2.1 Results Using the Probabilistic Choice Rule

Figure 5.1 presents objective and generational data for a product line with 6 products using the probabilistic choice rule with a random initialization. It can be clearly noted from this figure that the Lowest $n-1$ Share Crossover method offers objective improvement over the baseline scattered crossover. A smaller spread of data points is also provided by this crossover method, indicating a more effective means of reaching an optimum and the crossover operator's ability to counteract the inherent randomness within a GA.

The lower half of Figure 5.1 provides generational data concerning the same set of test trials, and again it is clearly evident that the Lowest $n-1$ Crossover method is computationally more efficient. Besides the two outlier points within the corresponding box plot, all remaining trials conducted by the crossover method were completed in a fewer number of generations than every trial completed with scattered crossover. These computational

benefits are reflected in other product line sizes, further validating the benefits in computational efficiency provided by the Lowest $n - 1$ Crossover method.

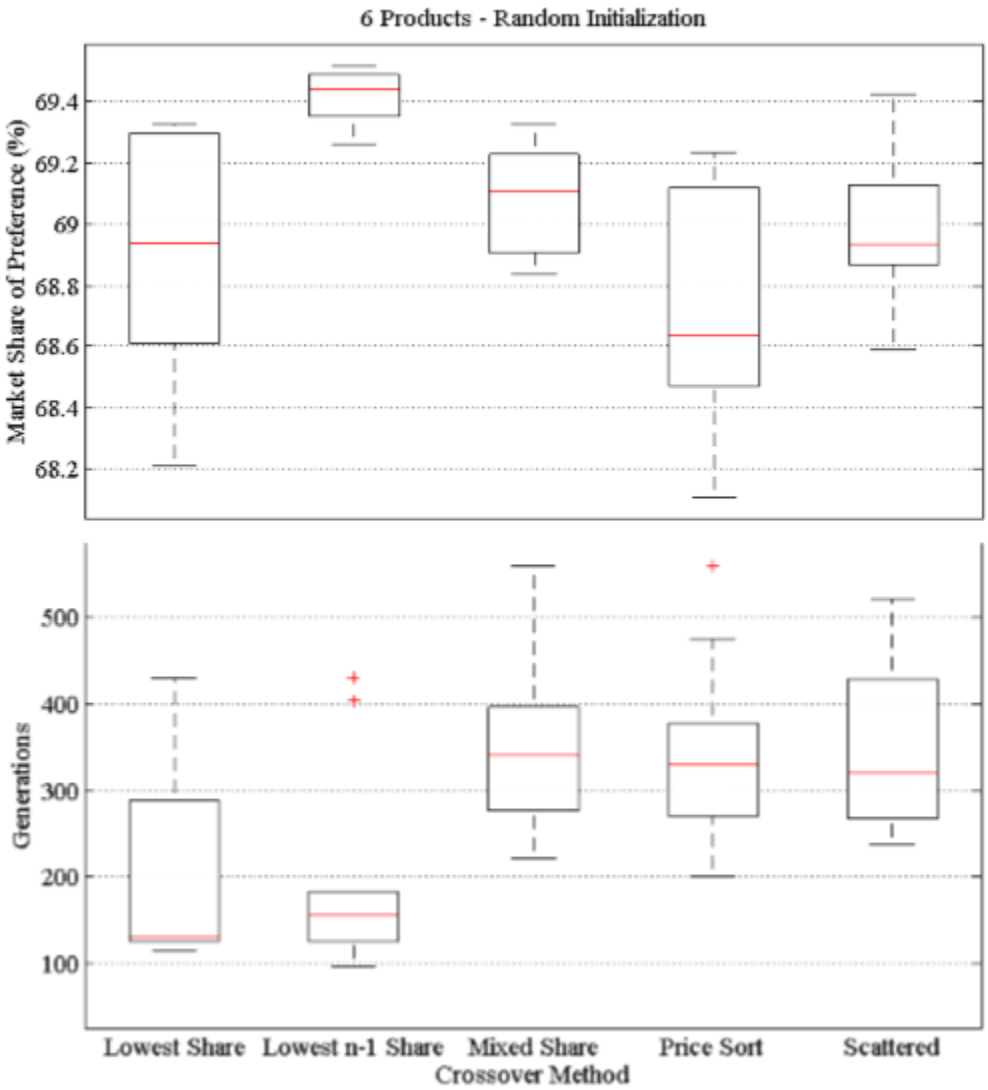

Figure 5.1: Objective and Generational Results for the Optimization of a Vehicle Product Line's Market Share with 6 Products Using the Probabilistic Choice Rule and Random Initialization

The data associated with the test trials that implemented a targeted initial population with the probabilistic choice rule are presented in Figure 5.2. Similar to the data presented in Figure 5.1, using the Lowest $n - 1$ Share Crossover operator demonstrates objective and generational improvement. However, these improvements are shown through the spread of data collected through the 10 trials. While the baseline scattered crossover can find solutions that achieve 70.3% market share, the box plot details a much larger spread of optimal performance over the 10 trials (a spread of 0.5 percentage points) when compared with the spread provided by the Lowest $n - 1$ Share Crossover (a spread of 0.25 percentage points). Analysis of the generational plot indicates even stronger advantages when using the Lowest $n - 1$ Share Crossover method. Ignoring the two outlier points on the box plot (located at roughly 150 and 390 generations), all trials conducted using the Lowest $n - 1$ Share Crossover reach convergence conditions within 100 generations. The baseline scattered crossover consistently reaches convergence within 300 generations (again, ignoring the two outlier points located at roughly 540 and 550 generations).

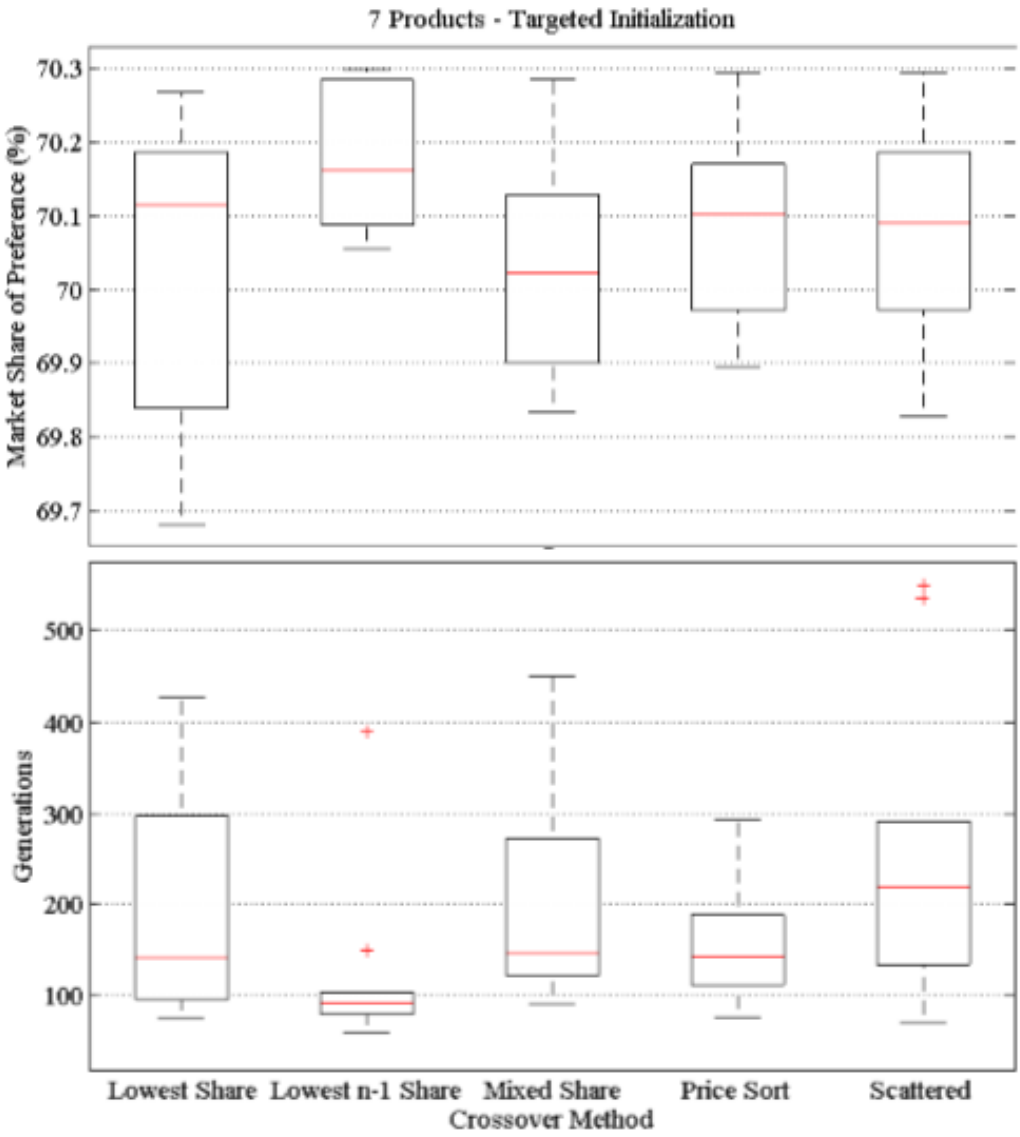

Figure 5.2: Objective and Generational Results for the Optimization of a Vehicle Product Line's Market Share with 7 Products Using the Probabilistic Choice Rule and Targeted Initialization

5.2.2 Results Using the First Choice Rule

The remaining trials are conducted using a first choice rule and are presented in Figure 5.3 and Figure 5.4 for a random initialization and targeted initialization, respectively. The results presented for both random and targeted initialization indicate significant objective improvements while still following the same previously mentioned trends highlighted by the probabilistic choice rule trials. The improvements are slightly less conclusive due to binary nature of the first choice rule, but are still conclusive nonetheless. The objective improvement provided by the Lowest $n - 1$ Share Crossover method shown in Figure 5.3 reveals that 50% of the conducted trials outperformed the trials conducted with scattered crossover.

These consistent improvements are also validated by generational improvements. Scattered crossover trials indicated a large spread of performance, ranging between 100 to 600 total generations to reach an optimum. Trials conducted with the Lowest $n - 1$ Share crossover reduced this range to 100 – 350 generations, providing more consistent guarantee that an optimum will be found much quicker. Interestingly, the Price Sort and Lowest Share crossover operators also demonstrate similarly strong generational improvements, but did not yield consistent objective improvements. Therefore, it is recommended that neither of these crossover methods be implemented for large-scale engineering design problems.

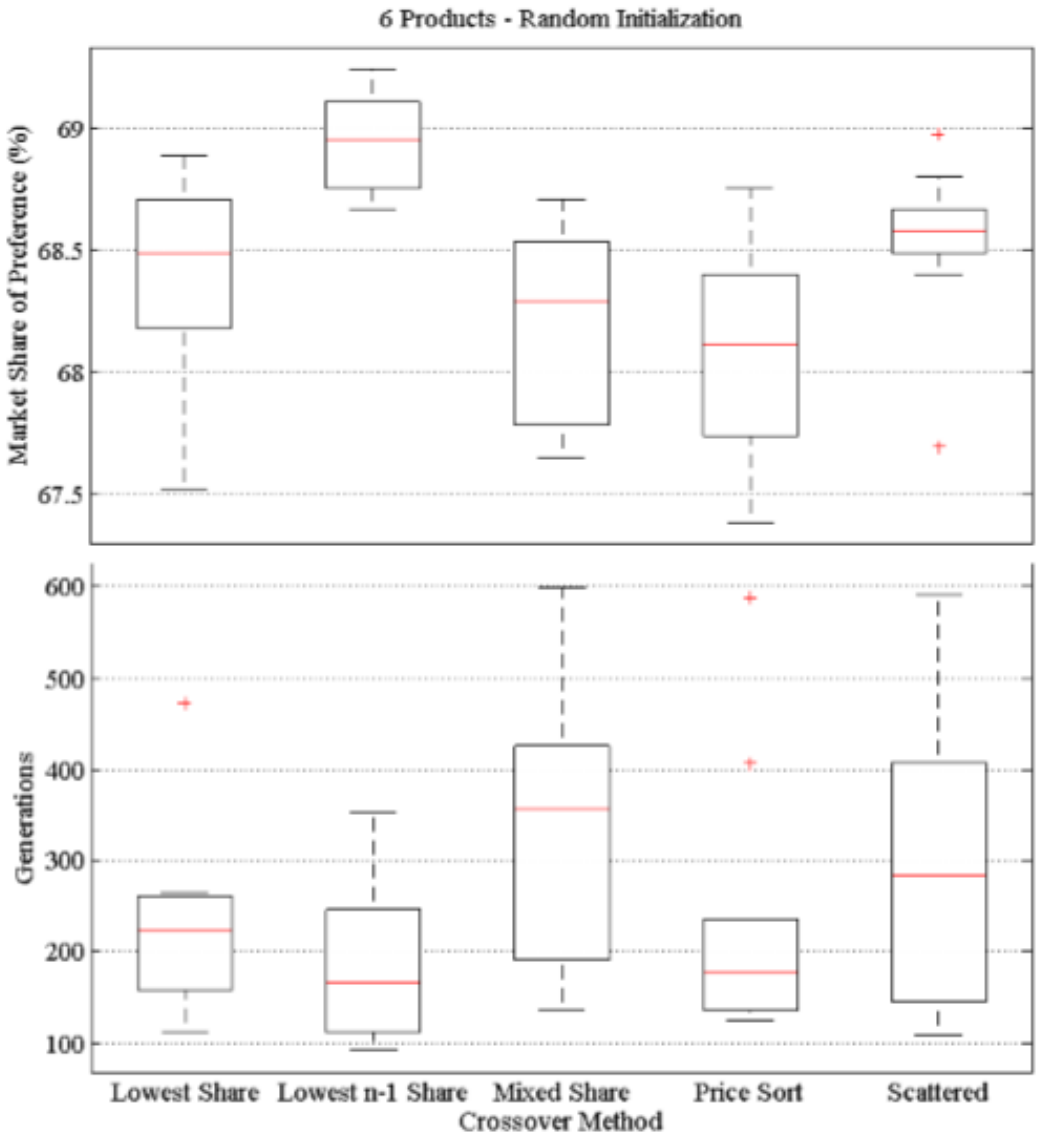

Figure 5.3: Objective and Generational Results for the Optimization of a Vehicle Product Line's Market Share with 6 Products Using the First Choice Rule and Random Initialization

Investigation of the targeted initialization trials conducted with the first choice rule (depicted in Figure 5.4) indicates a recurrence of the trends noted by Figure 5.1, Figure 5.2, and Figure 5.3. The Lowest $n - 1$ Share crossover operator again yields improved objective performance, increasing the algorithm effectiveness over the baseline scattered crossover. The lower half of Figure 5.4 again details the generational data collected from the test trials in question. Eliminating outlier data points, it can be seen that the Lowest $n - 1$ Share crossover method offers the same generational improvements as scattered crossover. When compared with the generational results in Figure 5.3, it can be seen that the improvements in computational efficiency can be largely accredited to the targeted population initialization.

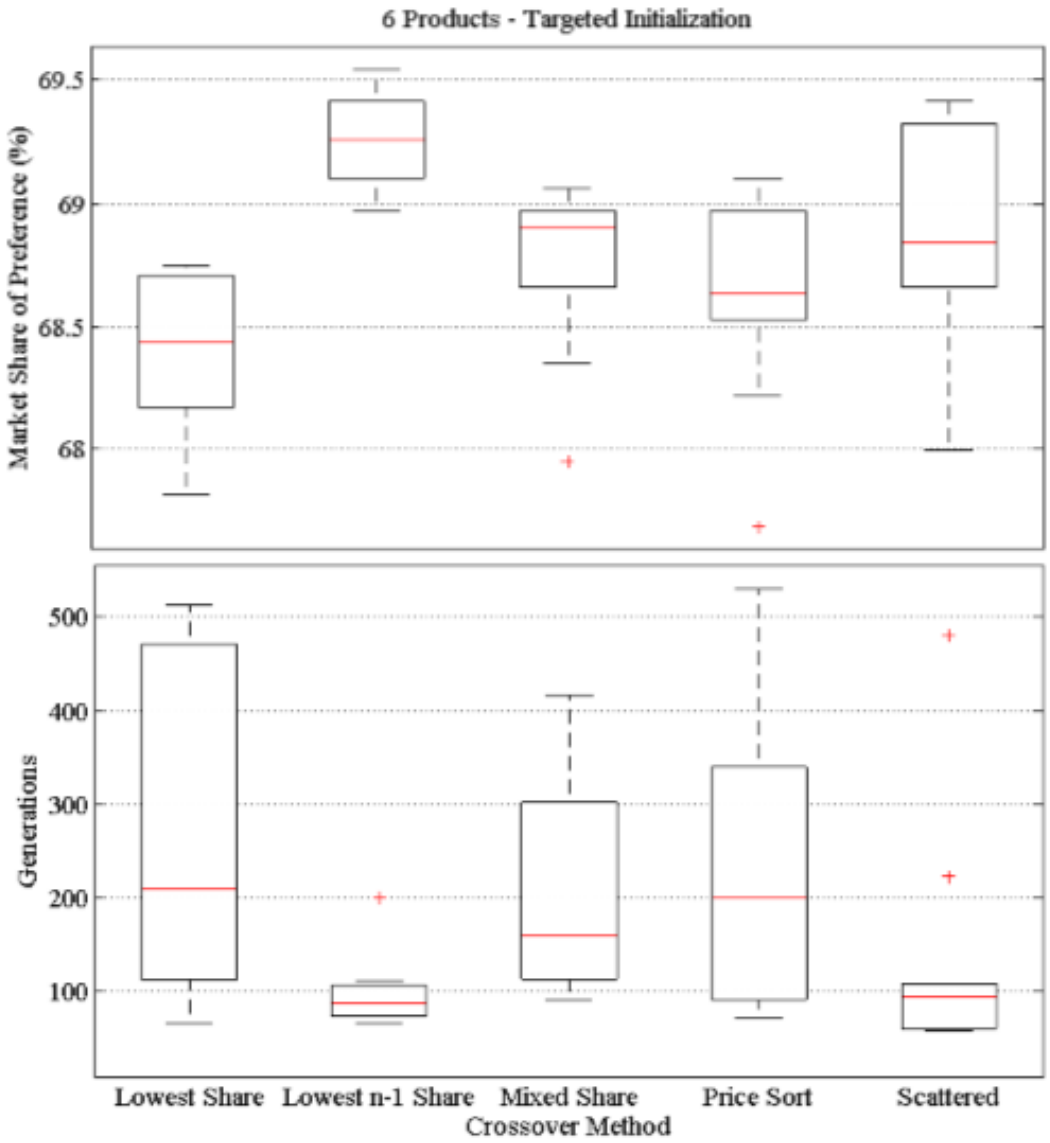

Figure 5.4: Objective and Generational Results for the Optimization of a Vehicle Product Line's Market Share with 6 Products Using the First Choice Rule and Targeted Initialization

Ultimately, the analysis of the results presented from the maximization of market share of preference indicates the strong computational and algorithmic benefits provided by the Lowest $n - 1$ Share crossover method when compared with both the baseline scattered crossover and with the other developed informed crossover operators. The following section will analyze the results from the maximization of the per capita contribution margin to determine if this recommendation holds for other objectives commonly used with product line optimization.

5.3 Maximization of Per Capita Contribution Margin of the Vehicle Problem

Contrary to previous sections, the analysis of the maximization of PCCM is subdivided by initialization method, so as to highlight the benefits provided by the informed crossover methods and separate these benefits from the ones offered by targeted initialization. Figure 5.5 depicts the test trial results conducted using a random initialization and a probabilistic choice rule, while Figure 5.6 depicts the test results conducted using a random initialization and a first choice rule. Following discussion of these two figures, Figure 5.7 and Figure 5.8 are presented, which depict the test trial results conducted using a targeted initialization for the probabilistic and first choice rules, respectively.

5.3.1 Results Using the Probabilistic Choice Rule

Analysis of Figure 5.5 again shows computational and generational benefits through the application of the Lowest $n-1$ Share crossover operator. The operator provides optimal objective results that encompass the spread yielded by scattered crossover. However, roughly half of the optimum objectives provided by the trials outperform the strongest objective data points provided by scattered crossover, indicating fairly consistent benefits. The lower half of Figure 5.5 also shows that roughly half of the trials conducted with the Lowest $n-1$ Share crossover yield generational improvement when compared to the strongest candidate from the scattered crossover trials. Interestingly, strong generational improvement is also provided by the Lowest Share crossover method, and some slight objective improvement is also offered by this operator. It should also be noted that the Price Sort crossover no longer indicates strong performance when used with the profit objective.

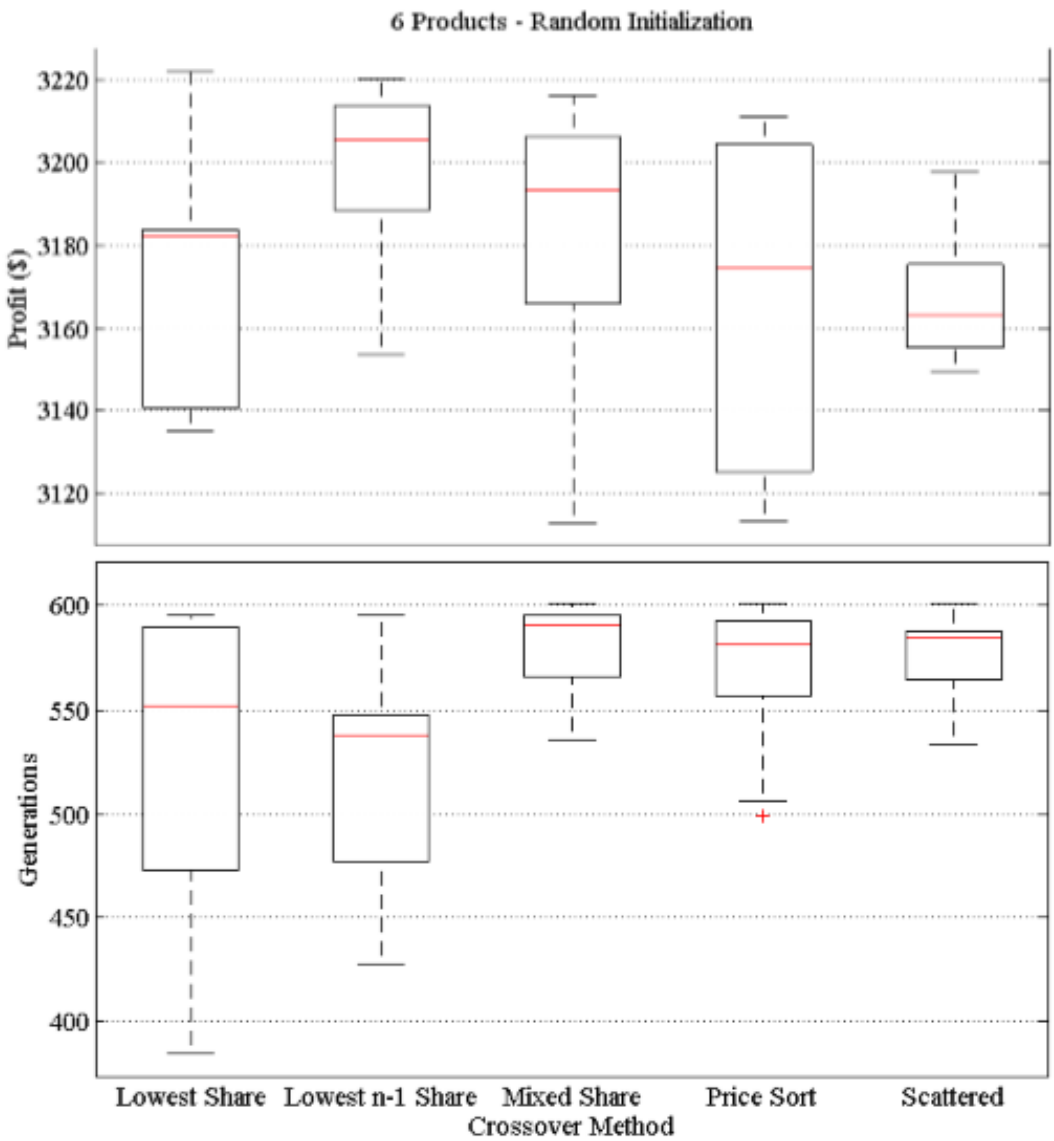

Figure 5.5: Objective and Generational Results for the Optimization of a Vehicle Product Line's Profit with 6 Products Using the Probabilistic Choice Rule and Random Initialization

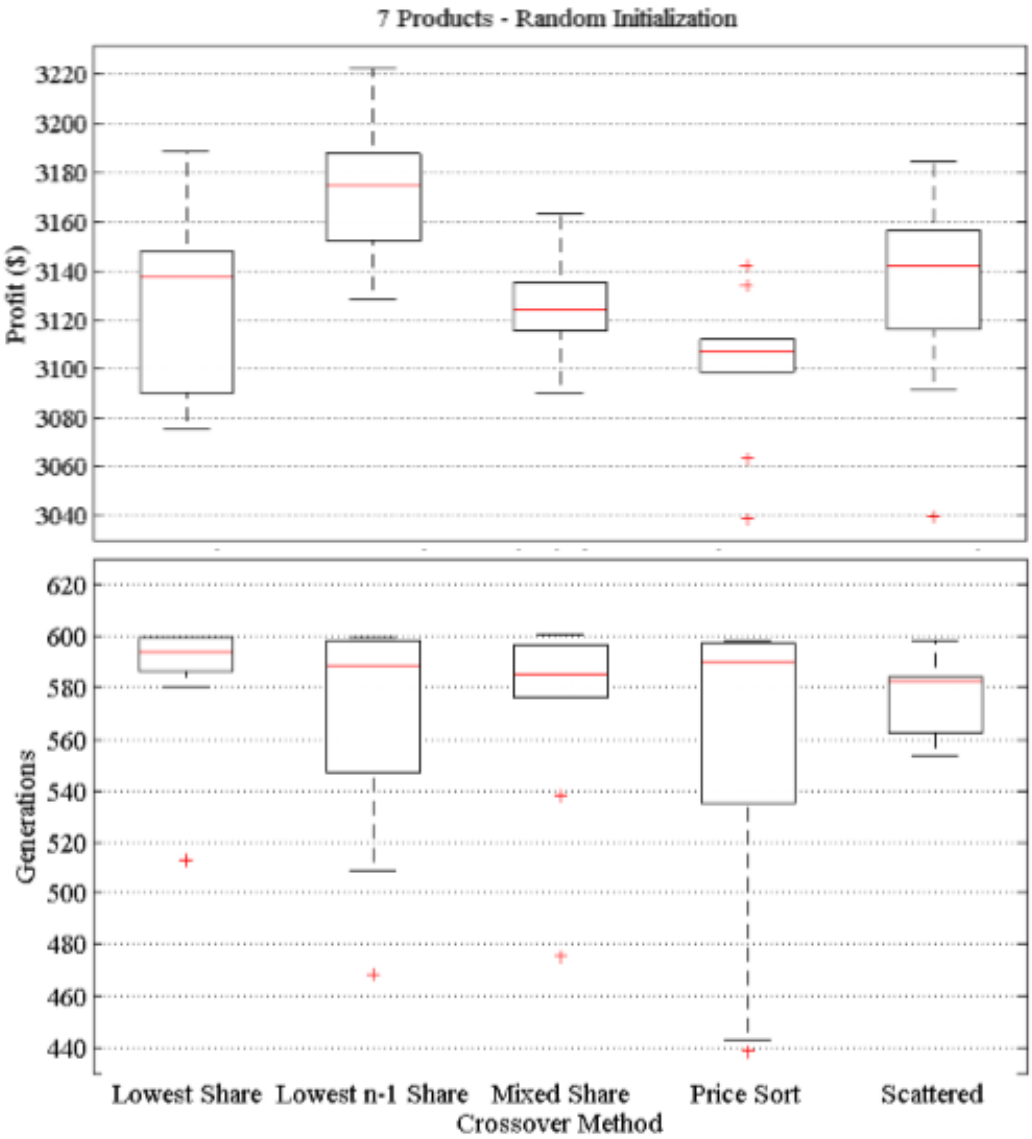

Figure 5.6: Objective and Generational Results for the Optimization of a Vehicle Product Line's Profit with 7 Products Using the First Choice Rule and Random Initialization

Figure 5.6 depicts the results from trials conducted with 7 products, a random initialization, and the first choice rule. The introduction of the first choice rule produces a more robust simulation space, theoretically yielding limited performance from trials. However, the trials conducted with the Lowest $n - 1$ Share crossover method still indicated capabilities of reaching the same optimal point discovered in the trials conducted with the probabilistic choice rule. The spread of these optimal points is ultimately driven downward, but optimal performance is still achieved by this crossover when compared with scattered crossover and the other developed informed operators. Notably, poor objective performance is displayed by the Price Sort crossover, behavior which is not expected considering the maximization of PCCM for the respective trials.

The lower half of Figure 5.6 also indicates generational maintenance achieved by the Lowest $n - 1$ Share crossover method when compared to the baseline trials. Behavior noted in some trials also hints at improved algorithm performance, as evidenced by the lower quartile of the box plot for this crossover method. Interestingly, the strongest crossover method for algorithm efficiency is the Price Sort crossover, implying that there may still be benefits to implementing pricing data when maximizing the per capita contribution margin of a product line.

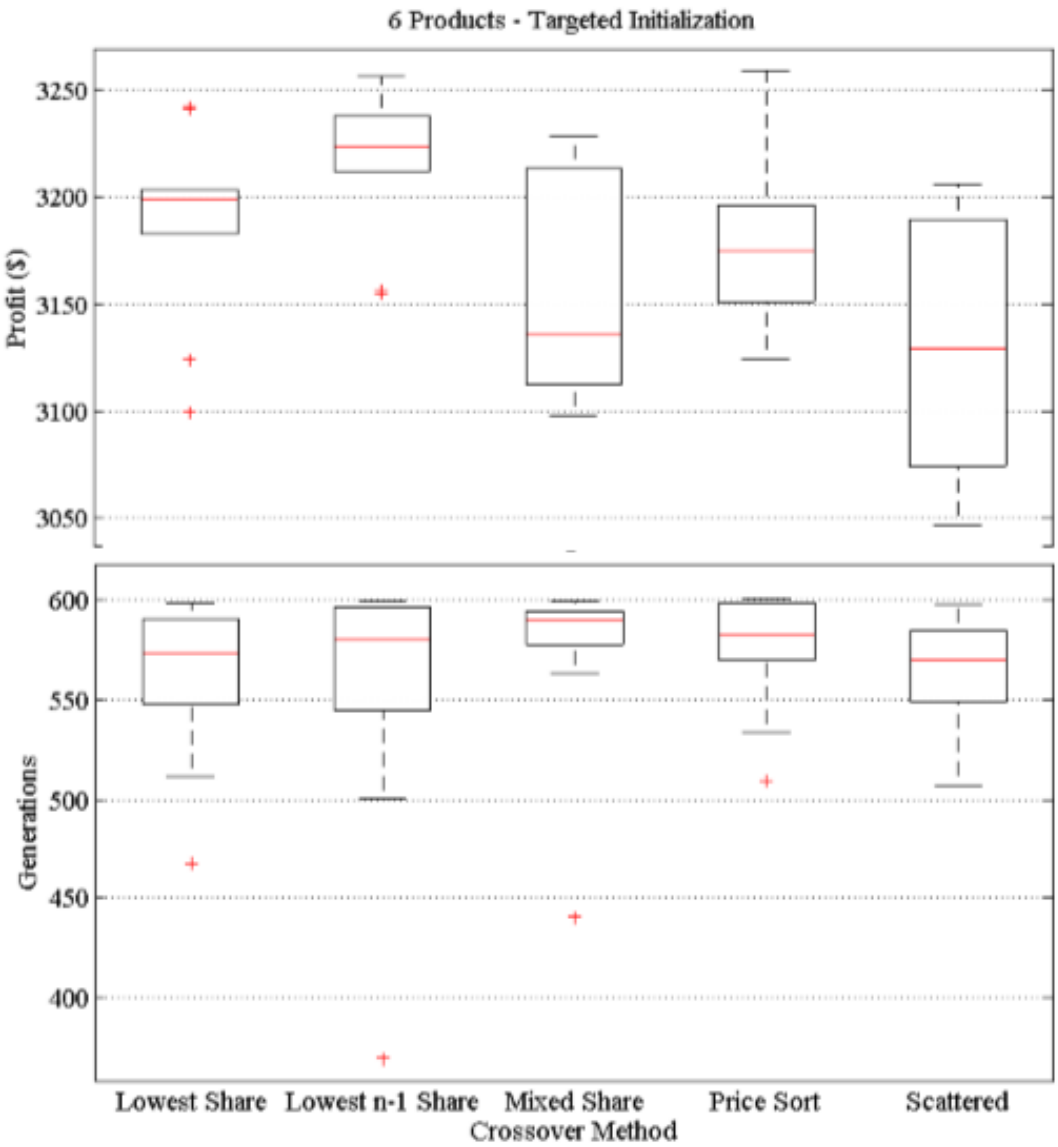

Figure 5.7: Objective and Generational Results for the Optimization of a Vehicle Product Line's Profit with 6 Products Using the Probabilistic Choice Rule and Targeted Initialization

5.3.2 Results Using the First Choice Rule

Figure 5.7 and Figure 5.8 present the results collected using targeted initialization with the probabilistic and first choice rules, respectively. The trends noted in these two figures are similar in nature and will be discussed concurrently. Both sets of test trials again indicate strong objective performance for trials conducted with the Lowest $n-1$ Share crossover operator when compared to scattered crossover trials. Interestingly, all of the informed operators show capabilities of outperforming scattered crossover, but the spread of the box plots depicting these trials also indicates unpredictability associated with these methods. This attribute is undesirable to a firm looking to consistently arrive at a product line configuration. It can also be noted that implementation of the first choice rule eliminates this data spread in all crossover operators except the Lowest $n-1$ Share crossover, also indicating the robustness of this informed operator.

The lower halves of Figure 5.7 and Figure 5.8 again detail generational results. It seems that the utilization of targeted initialization provides additional generational benefits that were initially noticed with only the Lowest $n-1$ Share crossover and random initialization. All tested crossover methods (informed and scattered) indicate a baseline of roughly 600 generations to reach an optimal design, indicating that more than likely the convergence criteria of 600 generations played some role in limiting the performance of the algorithm. Utilization of the first choice rule in Figure 5.7 shows a much larger spread of data points, indicating each crossover method's capability of arriving at an optimum in less generations.

Ultimately, all crossover operators used with a targeted initialization indicated comparable performance, leading one to refer to objective performance when analyzing these methods.

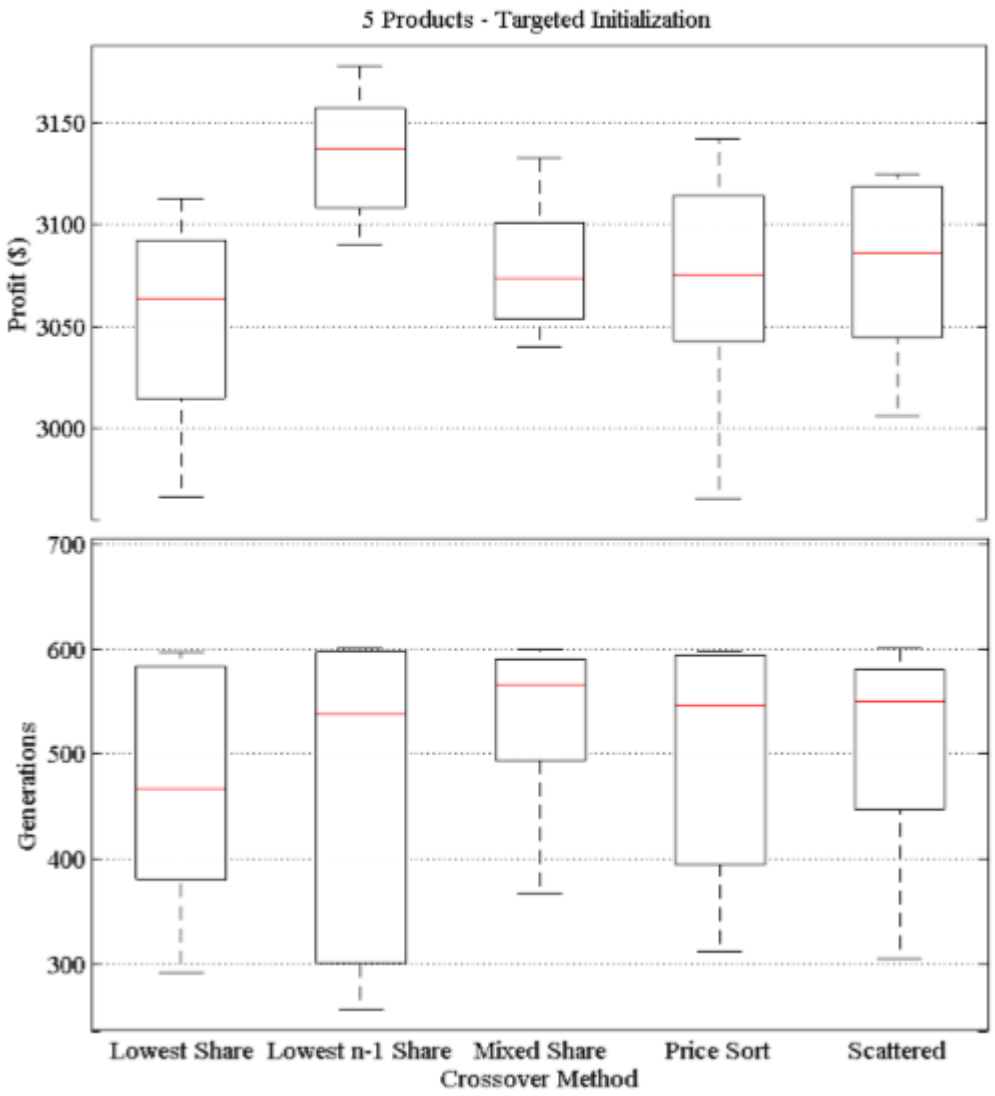

Figure 5.8: Objective and Generational Results for the Optimization of a Vehicle Product Line's Profit with 5 Products Using the First Choice Rule and Targeted Initialization

5.4 Chapter Summary

Overall analysis of the test trials to optimize a vehicle product line configuration yielded some notable trends. Most importantly, it is evident from the data collected that the Lowest $n-1$ Share crossover method outperforms the baseline scattered crossover and the other previously developed informed crossover operators when applied to a more complex and robust problem. This crossover method introduced significant objective improvements when maximizing both market share of preference and per capita contribution margin. In addition to these objective improvements, it was also noted that these gains were achieved in a fewer number of generations than other crossover methods (informed or scattered), indicating gains in computational efficiency. With no additional function calls necessary to run the Lowest $n-1$ Share crossover operator, the author highly recommends implementation of this operator when optimizing product lines with numerous attributes and implementing a single objective.

The experimental data and analysis presented in the following chapter will pertain to the efficacy of using the Lowest $n-1$ Share crossover method when optimizing more than one objective simultaneously. The trials conducted will again be compared against the baseline scattered crossover to determine the value of using the Lowest $n-1$ Share crossover method with multi-objective product line optimization problems.

CHAPTER 6 – MULTI-OBJECTIVE OPTIMIZATION

6.1 Multi-Objective Optimization Background

The past two chapters have focused on the benefits provided by informed crossover operators when applied to single-objective product line optimization problems, namely maximizing market share of preference (Equation (3.3)) and PCCM (Equation (3.4)), or profit. Ideally, one could identify a solution that maximizes both objectives simultaneously without sacrificing benefits provided by the optimal solutions to each objective problem. Previous research into multi-objective optimization problems has revealed that most often this is not possible [11,70–75]. Past work concerning targeted populations used the vehicle case study presented in this work and revealed the optimal solutions for both market share and profit maximization [10], pictured in Figure 6.1.

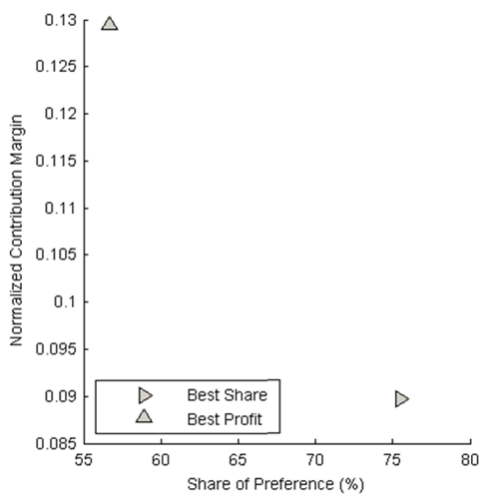

Figure 6.1: Performance of Optimal Share Configuration and Optimal PCCM Configuration for Vehicle Case Study [10]

Due to the lack of a single optimal solution within the design space, the designer is left with a tradeoff decision. Many firms typically drive their decisions based on profit maximization, but there are instances when firms will sacrifice profit in exchange for increased market share to establish a footprint within a marketplace. Between the two points noted in Figure 6.1 are a collection of non-dominated points denoted a Pareto Frontier [76]. The Pareto Frontier defines a full set of solutions over the range of objective function values, providing designers with a collection of designs to consider when faced with a tradeoff decision [77]. The construction of a Pareto Frontier requires a multi-objective optimization approach.

The simplest means of solving a multi-objective optimization problem is through a weighted sum approach [78], which establishes combinations of weights for each single-objective to reduce a multi-objective problem to a single objective one. However, a weighted sum approach does not generate solutions in the non-convex region of the Pareto Frontier, cannot create an even spread of solutions, and is computationally expensive when compared with other approaches [78]. Conversely, this research uses a multi-objective genetic algorithm (MOGA) to search for product line configurations. The NSGA-II algorithm [57] is a commonly used MOGA and is used for this research. The defining difference between a standard GA and a MOGA is the ability of a MOGA to determine which solutions are dominated by others. The GA settings discussed in Section 2.4 are used and combined with a non-dominated sorting algorithm to establish the MOGA used in this research.

To quantify and analyze the solution quality produced from the MOGA, a hypervolume metric is adopted [79]. The hypervolume metric captures both the distance and spread of the

frontier in a single value, making it an effective metric for determining solution quality in a space. A larger hypervolume value corresponds to a better frontier, or set of non-dominated points. The implementation of a hypervolume metric requires that all objectives be normalized before analysis, so that one objective isn't unfairly weighted over another. For the purposes of this research, the PCCM objective is normalized to be on a 0 to 1 scale, so as to match the scale used by the market share of preference (the true values are backed out for analysis after completion of the optimization trials).

When analyzing a two-objective case study (as is the case with this problem) the hypervolume corresponds to the area underneath the curve in reference to the origin. It should be noted that this area calculation is only valid if both objectives have lower bounds of 0. Figure 6.2 depicts an example of this using the frontier in Figure 6.1 [10].

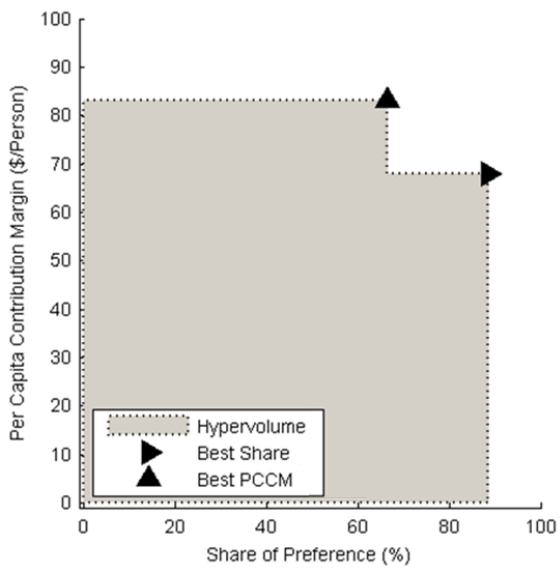

Figure 6.2: Example Hypervolume for MP3 Case Study [10]

6.2 Experimental Procedure for Multi-Objective Optimization

To conduct an analysis of the benefits of implementing an informed crossover with a MOGA, only the Lowest $n-1$ Share crossover will be applied (due to the promising results it displayed in Chapter 5). The full-factorial design outlined in Section 3.4 is applied to the multi-objective experiment with some minor alterations. Due to the relatively simplistic nature of the problem, the MP3 product line optimization problem is not explored with the MOGA. The number of trials conducted for each experimental setup is reduced to 5 trials (as opposed to the previously used 10 trials). The 600 generation convergence criterion is maintained, but due to the nature of the multi-objective problem, generational information is not presented (as each point on the frontier would have generational data associated with it, yielding an inordinate amount of points). The population size is initially two times the size of the design string and expands as more non-dominated designs are discovered.

The goal of each optimization trial is to maximize both the market share of preference and PCCM. The problem formulation can be seen in Equation (6.1) [10].

$$\begin{aligned}
&\textbf{Maximize:}\ \text{Share of Preference}\\
&\text{Per Capita Contribution Margin}\\
&\textbf{Change:}\ \text{Feature levels } (X_{jkl})\\
&\textbf{Subject To:}\ \text{Price level bounds}\\
&\text{No identical product variants in a line}\\
&\text{Feature level bounds}
\end{aligned} \tag{6.1}$$

Having established an experimental setup, the following sections will discuss the results of these test trials. In contrast to past data analyses, the collected data will be analyzed using the hypervolume metric and a visual inspection of the developed Pareto frontier.

6.3 Experimental Results from MOGA Trials

The following subsections are divided by market model for analysis of the collected hypervolume metrics, followed by a visual analysis of the generated Pareto frontiers. A full enumeration of all generated frontiers is presented in Appendix B for reference.

6.3.1 Multi-Objective Optimization Using the Probabilistic Choice Rule

Figure 6.3 presents the hypervolume spread collected across the five trials conducted for each product line size (i.e. 4, 5, 6, and 7 products) using a random initialization. The implementation of the Lowest $n-1$ Share crossover operator yielded an average percent increase in hypervolume of 2.334%. Inspection of the data collected from the trials reveals the performance improvements offered by the Lowest $n-1$ Share crossover method. A 95% confidence interval around the mean hypervolume is presented in Table 6.1. The calculated confidence intervals have very little overlap, indicating consistent improvement offered by the Lowest $n-1$ Share crossover when compared to the baseline scattered crossover.

Table 6.1: 95% Confidence Interval around Mean Hypervolume for Random Initialization Multi-Objective Trials Using the Probabilistic Choice Rule

		Crossover Method	
		Lowest $n-1$ Share	**Scattered**
Product Line Size	**4 Products**	1797.3 ± 26.0	1770.1 ± 20.1
	5 Products	1865.2 ± 19.8	1816.5 ± 17.3
	6 Products	1901.7 ± 16.3	1842.6 ± 15.7
	7 Products	1919.8 ± 14.8	1883.9 ± 25.2

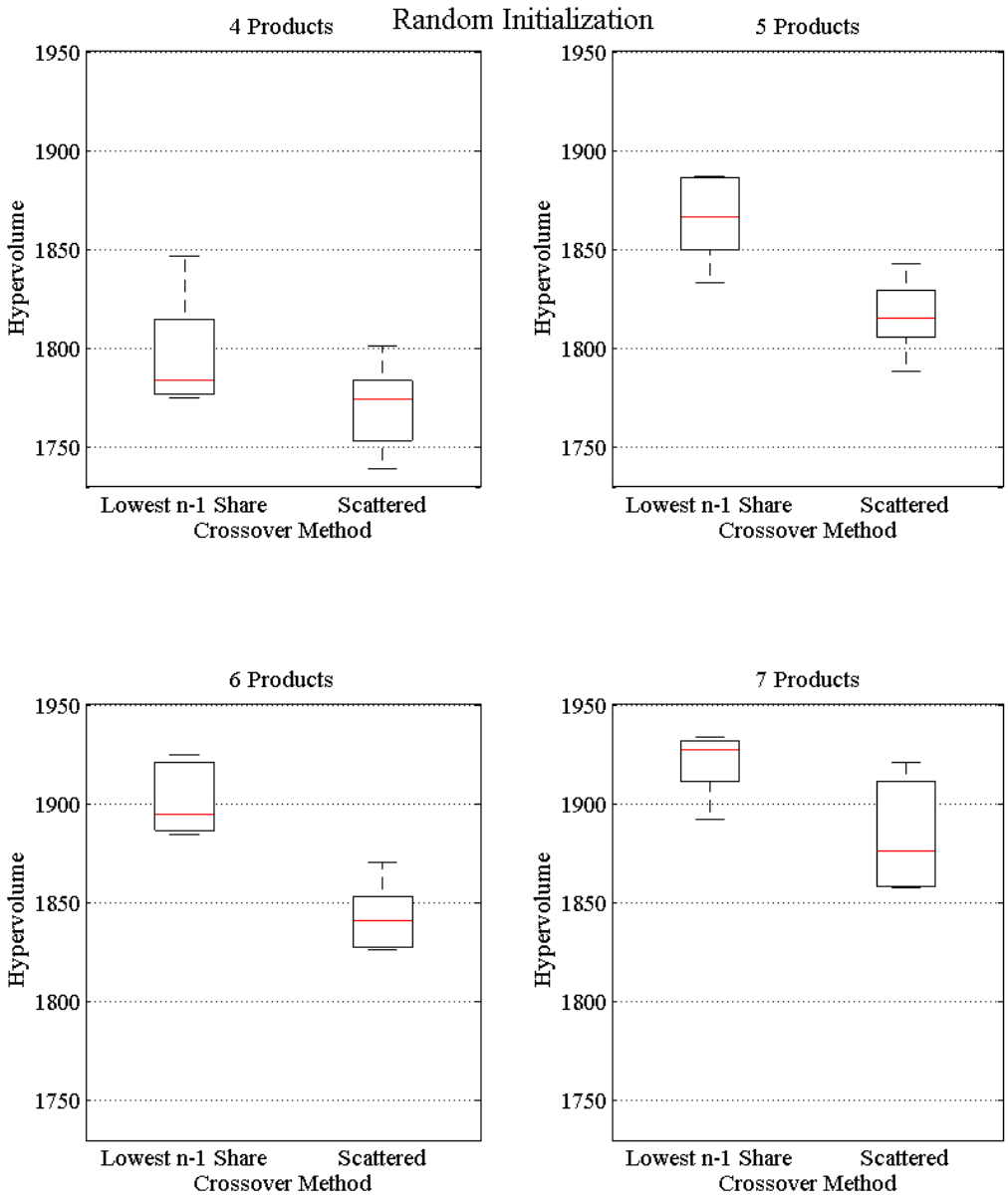

Figure 6.3: Hypervolume Data for Trials Conducted Using Random Initialization and the Probabilistic Choice Rule

The results from the five trials conducted using a targeted initialization method and the probabilistic choice rule are presented in Figure 6.4. Product lines with 5 and 7 products yield improved hypervolume values when optimized using the Lowest $n-1$ Share crossover method. However, lines with 4 products seem to render no benefit from implementation of the informed operator, and lines with 6 products seem to occasionally yield richer solution spaces when scattered crossover is used. These discrepancies can be due to differences within the design space as the problem formulation changes. Another potential reason could be due to the inherent randomness associated with genetic algorithms. This explanation can be further defended by the relatively small confidence interval spread noted by the optimization trials using the Lowest $n-1$ Share crossover method with 6 product optimization trials, as show on Table 6.2. Ultimately, implementation of targeted initialization within these trials indicates that the Lowest $n-1$ Share crossover may not offer significant hypervolume benefits when compared to initializing with a targeted population.

Table 6.2: 95% Confidence Interval around Mean Hypervolume for Targeted Initialization Multi-Objective Trials Using the Probabilistic Choice Rule

		Crossover Method	
		Lowest $n-1$ Share	**Scattered**
Product Line Size	**4 Products**	1789.5 ± 22.1	1797.4 ± 16.8
	5 Products	1858.1 ± 31.4	1833.6 ± 23.0
	6 Products	1882.3 ± 5.5	1893.5 ± 34.8
	7 Products	1922.3 ± 21.8	1889.5 ± 13.2

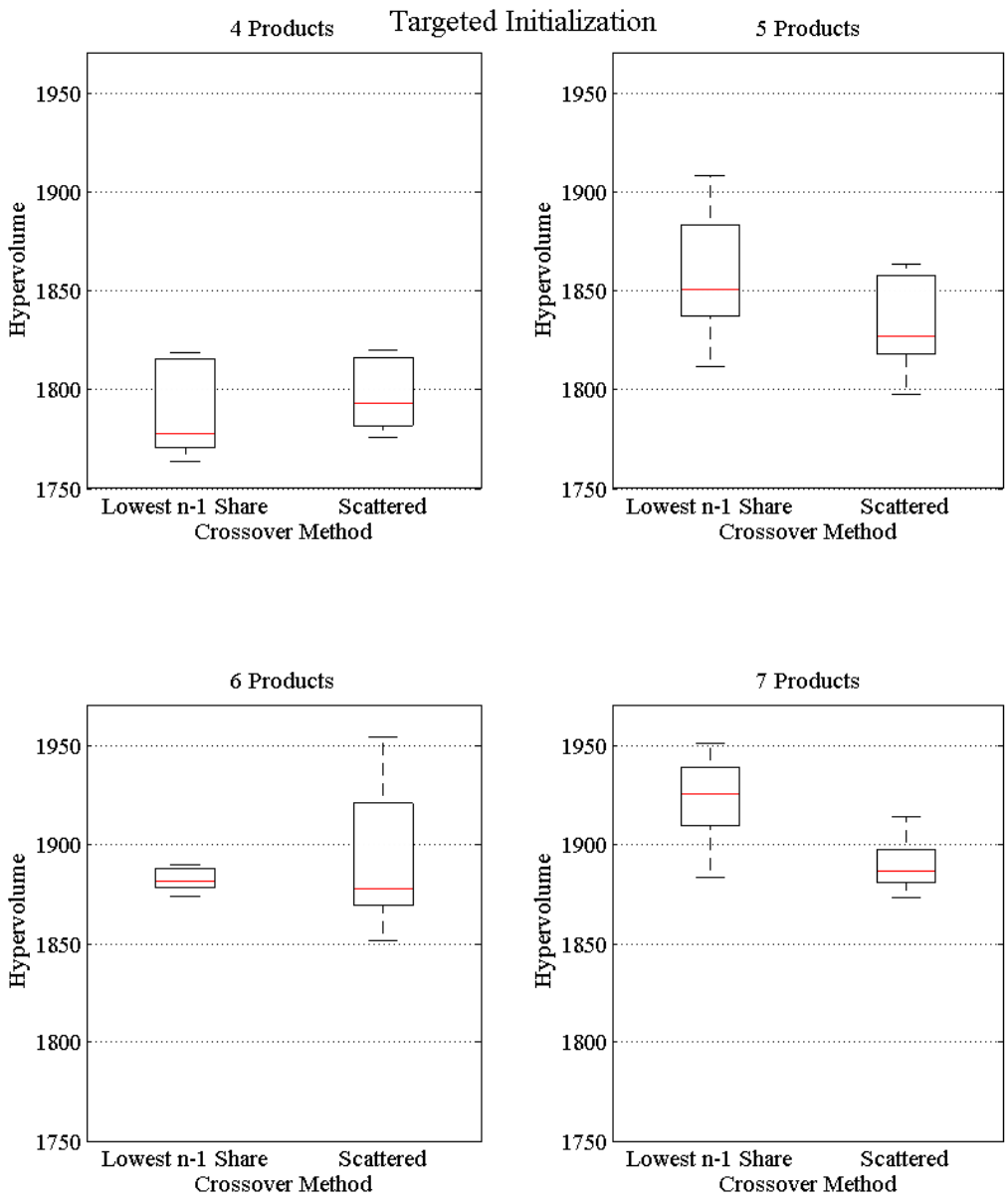

Figure 6.4: Hypervolume Data for Trials Conducted Using Targeted Initialization and the Probabilistic Choice Rule

6.3.2 Multi-Objective Optimization Using the First Choice Rule

The data presented in Figure 6.5 and Table 6.3 was collected through test trials conducted using the first choice rule with a random initialization technique. As seen with the trials conducted using a random initialization with the probabilistic choice rule, the Lowest $n-1$ Share crossover method offers a solution space with a larger hypervolume, indicating the informed crossover operator provides a more even and tighter frontier of solutions for the designer to select from. It should be noted that the interval ranges presented in Table 6.3 are slightly larger than the ranges calculated using the same experimental setup with the probabilistic choice rule. This is likely due to the robustness of the first choice rule and showcases the informed crossover operator's ability to handle the more complicated design space.

Table 6.3: 95% Confidence Interval around Mean Hypervolume for Random Initialization Multi-Objective Trials Using the First Choice Rule

		Crossover Method	
		Lowest $n-1$ Share	**Scattered**
Product Line Size	**4 Products**	1766.3 ± 35.5	1733.2 ± 22.0
	5 Products	1800.2 ± 23.7	1751.7 ± 21.7
	6 Products	1830.2 ± 24.9	1807.3 ± 19.6
	7 Products	1853.5 ± 23.3	1792.9 ± 17.7

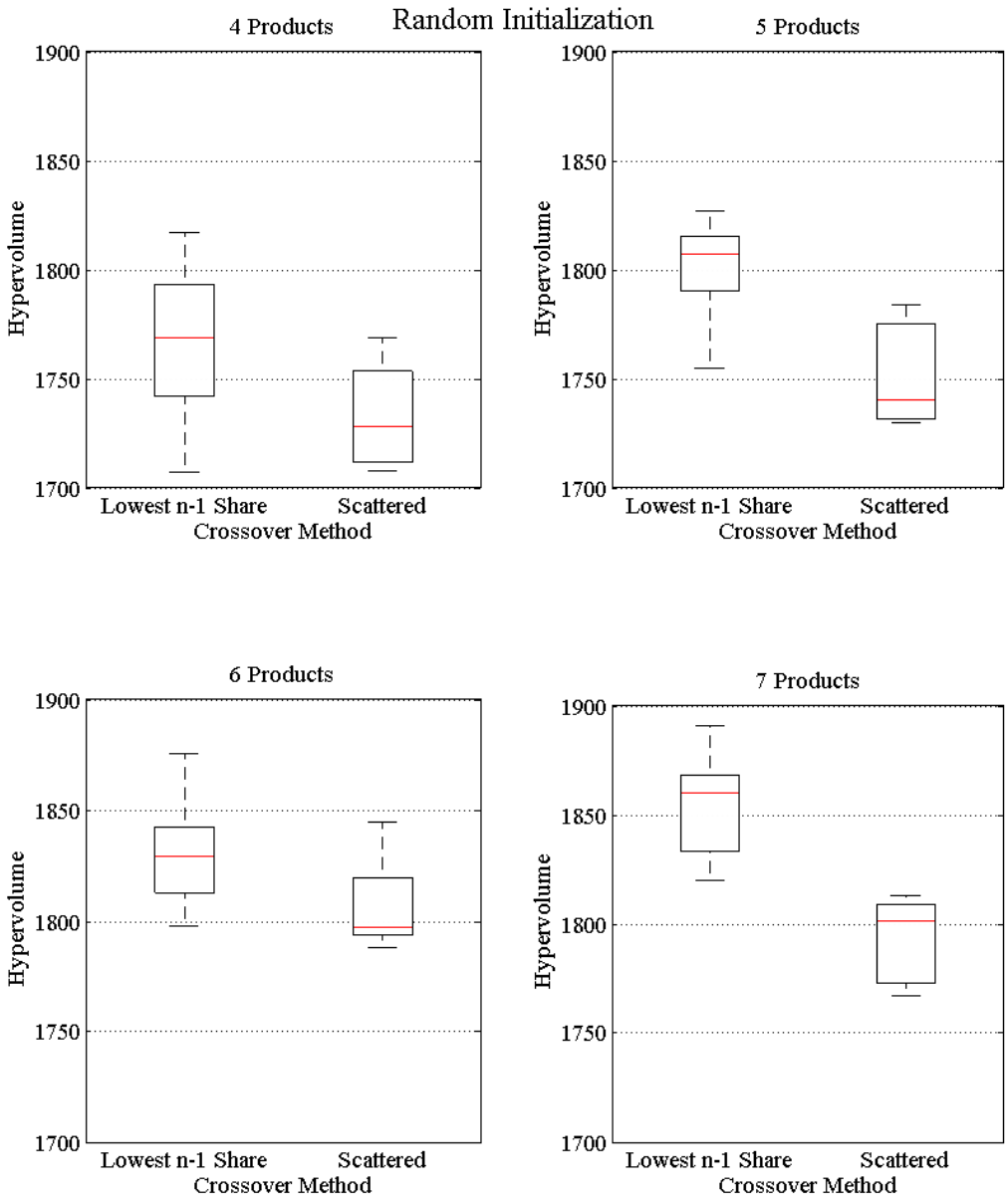

Figure 6.5: Hypervolume Data for Trials Conducted Using Random Initialization and the First Choice Rule

Table 6.4 and Figure 6.6 present a summary of the experimental results conducted using a targeted initialization using the first choice rule. Similar to the results presented from the test trials conducted with a targeted initialization and the probabilistic choice rule, the informed Lowest $n-1$ Share crossover method appears to show limited to no improvement when used with a targeted population. In fact, the 95% confidence interval created for product lines with 7 products show an overall improvement through the implementation of scattered crossover, indicating the Lowest $n-1$ Share crossover reached a false optimum, potentially stalling during the optimization along a false frontier.

Table 6.4: 95% Confidence Interval around Mean Hypervolume for Targeted Initialization Multi-Objective Trials Using the First Choice Rule

		Crossover Method	
		Lowest $n-1$ Share	**Scattered**
Product Line Size	**4 Products**	1763.3 ± 11.8	1741.6 ± 27.6
	5 Products	1805.0 ± 24.6	1768.9 ± 12.3
	6 Products	1827.6 ± 24.5	1802.2 ± 19.9
	7 Products	1859.9 ± 23.3	1849.0 ± 14.9

Generally, some hypervolume improvements are still shown through the use of the informed crossover operator. These improvements have shown limitations when combined with an informed initialization operator or a more robust market model when compared to standard crossover operators. A further analysis of the actual frontiers produced through test trials is needed to fully determine the benefits provided by the informed crossover operator.

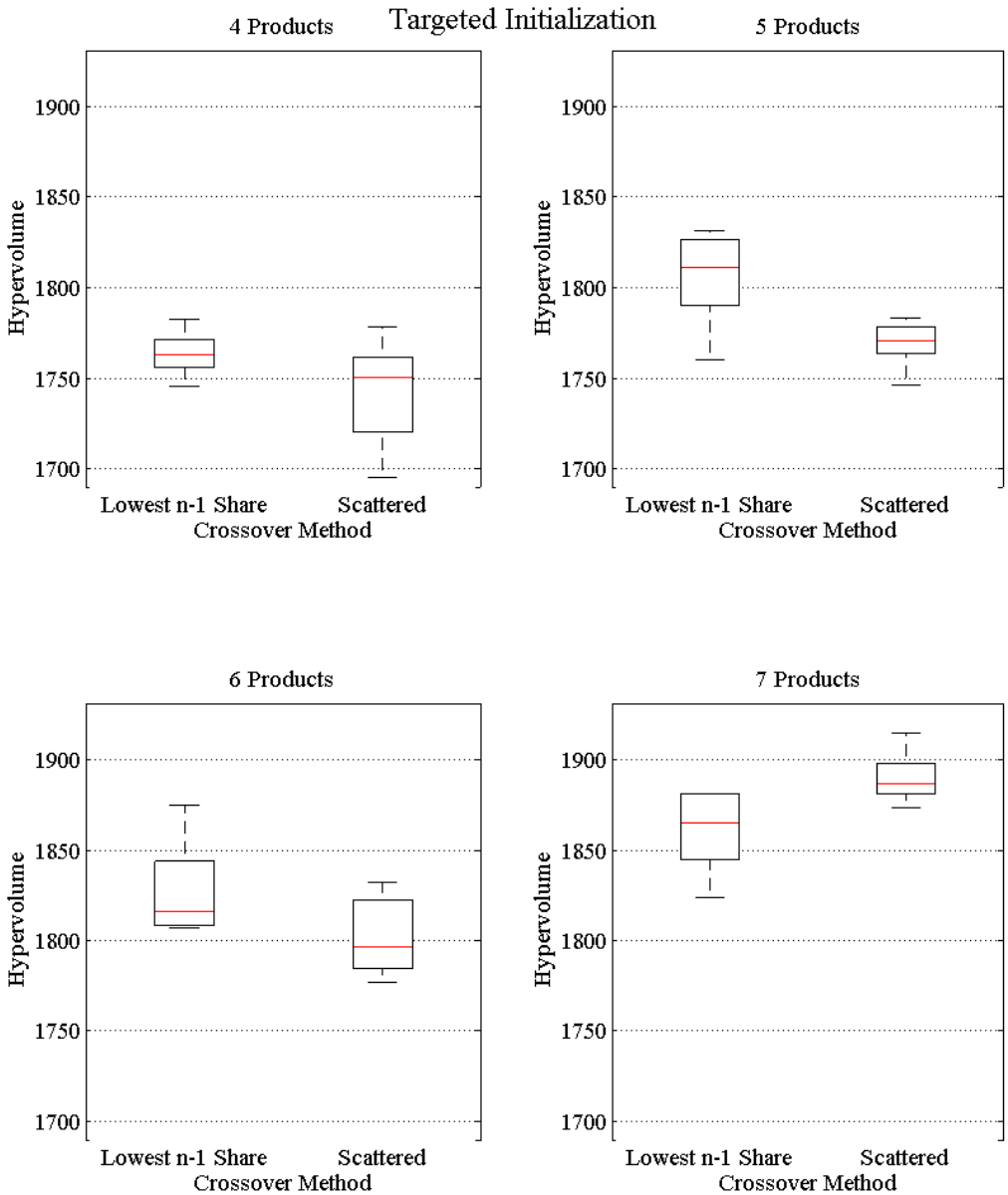

Figure 6.6: Hypervolume Data for Trials Conducted Using Targeted Initialization and the First Choice Rule

6.3.3 Pareto Frontier Analysis

To analyze the effects of implementing an informed crossover method (specifically, the Lowest $n-1$ Share crossover operator), a standard product line size of 7 products was selected for comparison. The following discussion highlights the test trials conducted with the aforementioned product line size with variations in population initialization and market model used in the optimization algorithm. The frontiers displayed include the full collection of all five frontiers created through the five trials conducted through the experimental design, so as to discover any trends created through the different trials. A numerical analysis of the collected hypervolume metrics was completed in subsections 6.3.1 and 6.3.2. The analysis included in this subsection is comprised of a visual inspection of the generated frontiers.

Figure 6.7 and Figure 6.8 present the results from multi-objective optimizations conducted with a random initialization and the probabilistic and first choice rules, respectively. Visual inspection of both of these figures reveals interesting trends within the frontiers. Within the sections of the frontiers with lower market shares, it appears as if performance from both crossover methods is generally comparable. As the market share of preference increases, noticeable improvements are provided by the Lowest $n-1$ Share crossover method. The generated frontiers expand outwards in these regions, indicating improved performance in areas with higher market share. It can be inferred that the expansion of the frontiers in this area is due to the fact that the informed crossover method being tested leverages products within the product line with higher shares, theoretically boosting the market share of the whole line. Another interesting trend to note is the “knee” in the Pareto frontier presented in

Figure 6.8. This indicates a trend towards an optimal solution (as opposed to a frontier of solutions) and showcases the frontier strength offered by the Lowest $n-1$ Share crossover method.

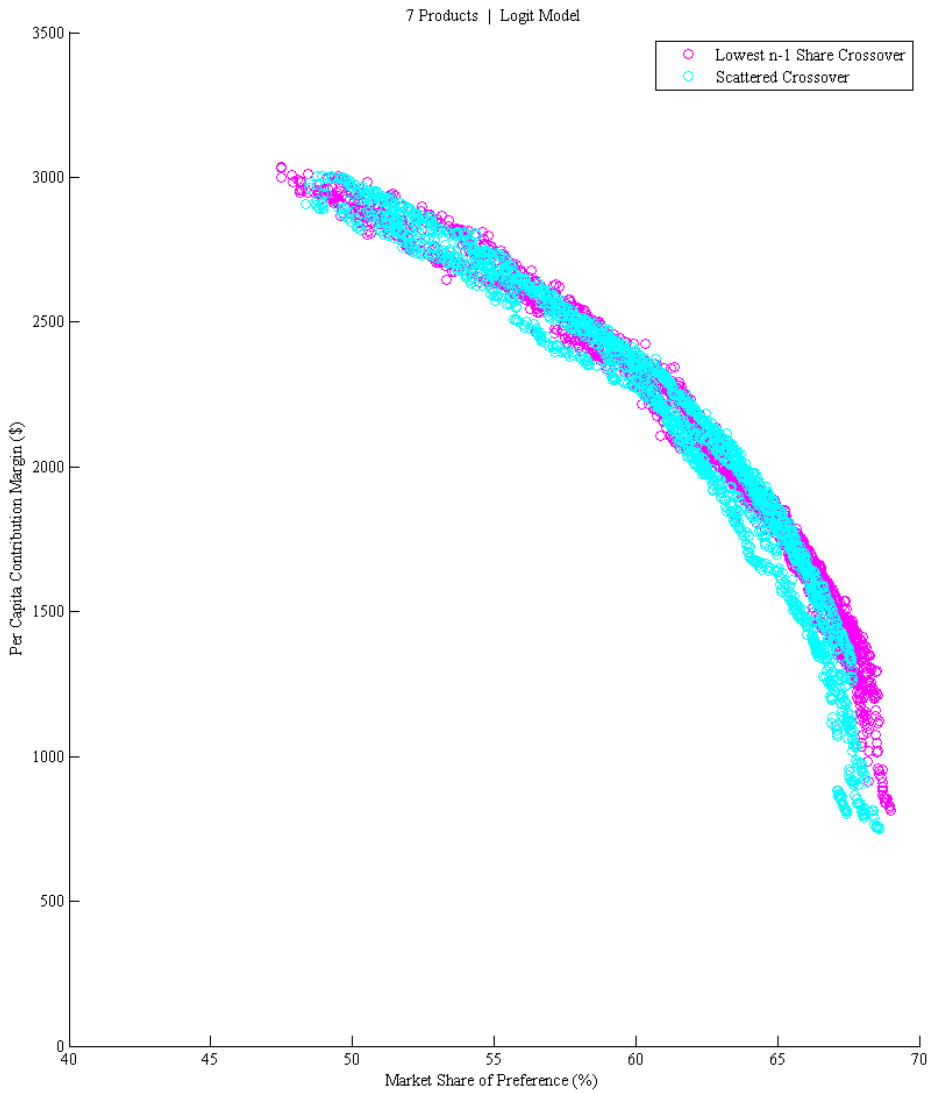

Figure 6.7: Pareto Frontiers for Test Trials Conducted with the Probabilistic Choice Rule and a Random Initialization

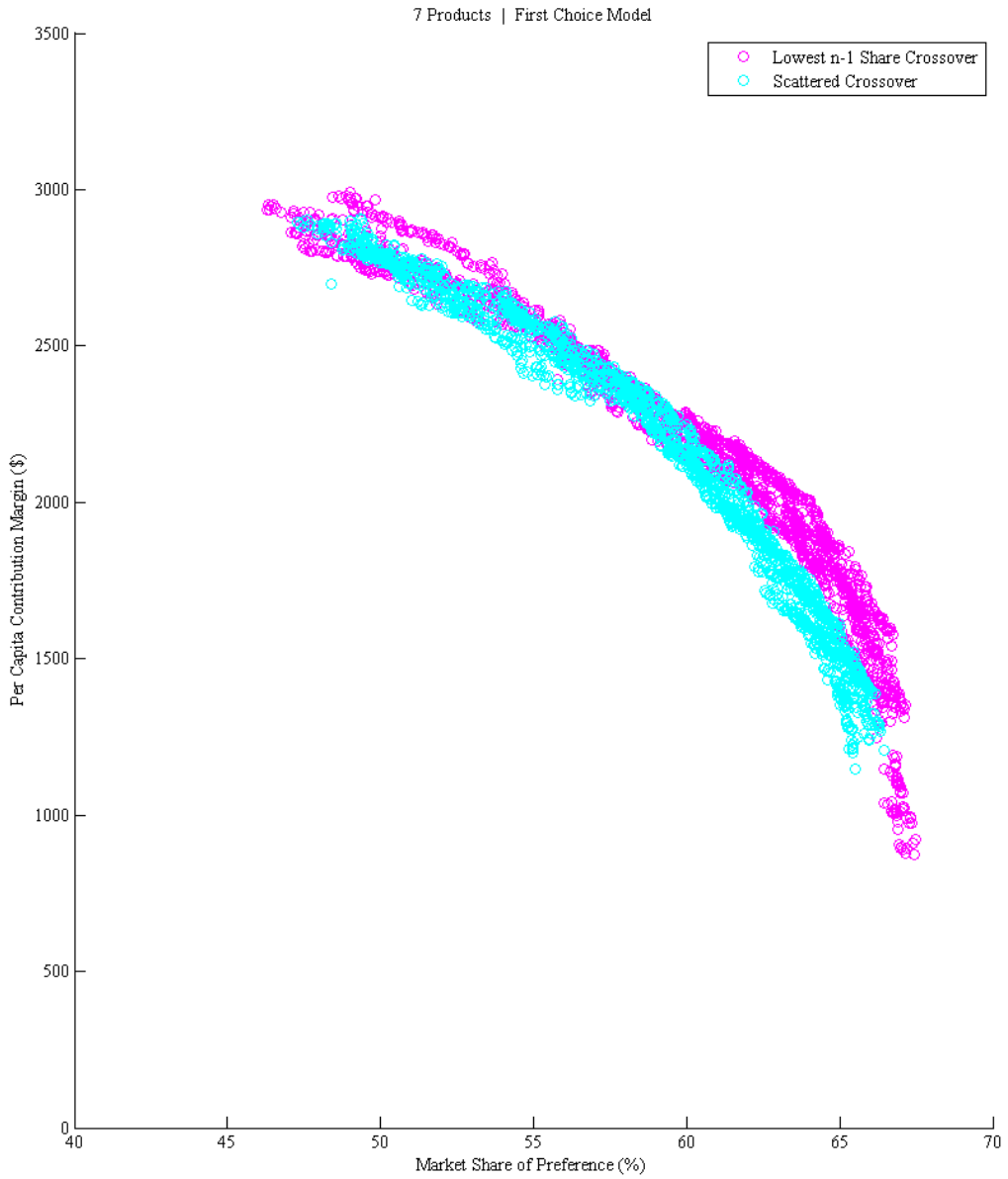

Figure 6.8: Pareto Frontiers for Test Trials Conducted with the First Choice Rule and a Random Initialization

The implementation of a targeted initialization significantly improved the performance of the trials conducted with scattered crossover, as evidenced by Figure 6.9 and Figure 6.10. These figures indicate only slight improvements of the performance of trials run with the Lowest $n-1$ Share crossover. Combining the visual analysis of these frontiers with the hypervolume data depicted in Figure 6.4 and Figure 6.6 indicates that using a targeted initialization method with an informed crossover operator does not aid the optimization of complex product lines significantly. Using one method or the other (with a robust market model such as the first choice rule) will produce comparable results. The improvements in the frontier provided by the scattered crossover method in Figure 6.10 are attributed to a sup-optimum present within the design space that the Lowest $n-1$ Share crossover is driven to during the optimization routine. It is theorized that these optimal products are generated in the targeted initialization and are thus not altered during the optimization routine.

Overall analysis of the Lowest $n-1$ Share crossover method when used in multi-objective optimization trials of product lines indicates higher hypervolume values and expanded Pareto frontiers within the generated solution spaces. The hypervolume and frontier benefits generally offered by this method provide significant improvements over results generated through standard crossover method driven test trials. Ultimately, the author strongly suggests implementation of this crossover method (or an informed initialization method) when optimizing complex product lines.

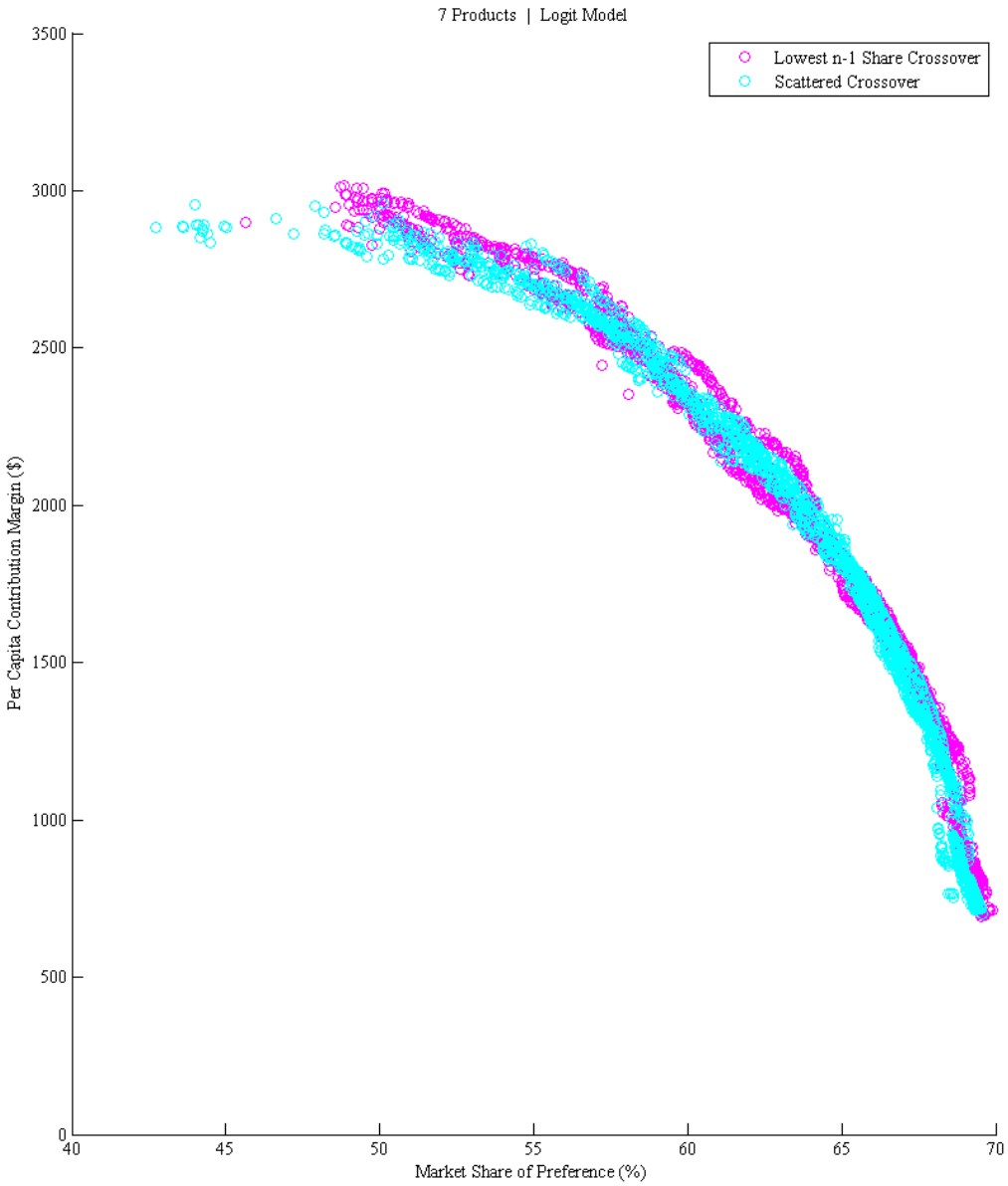

Figure 6.9: Pareto Frontiers for Test Trials Conducted with the Probabilistic Choice Rule and a Targeted Initialization

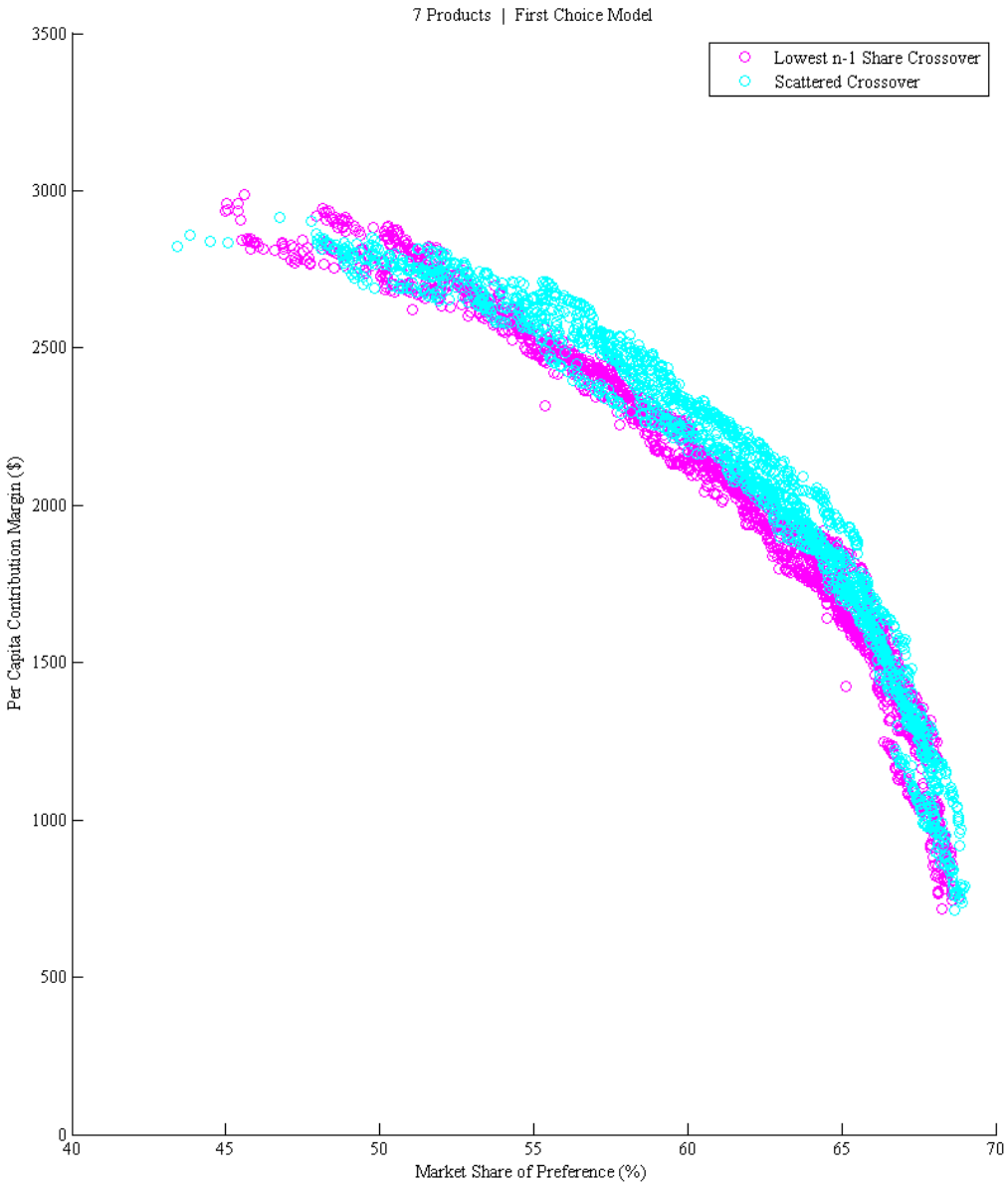

Figure 6.10: Pareto Frontiers for Test Trials Conducted with the First Choice Rule and a Targeted Initialization

CHAPTER 7 – SUMMARY AND CONCLUSIONS

7.1 Research Overview

The research conducted in this thesis follows the structure presented in Figure 7.1. The Creation phase (covered in Chapter 3) encompasses the development of the informed crossover operators to be tested were formally developed. Following the creation of these operators, the Hypothesis Testing phase of the study was entered. Within this phase (covered in Chapter 4), the informed crossover operators were validated with a relatively simple product line development optimization problem (the single objective MP3 product line problem). This phase served to ensure that the informed crossover operators matched standard crossover performance before testing the operators on a more complex problem.

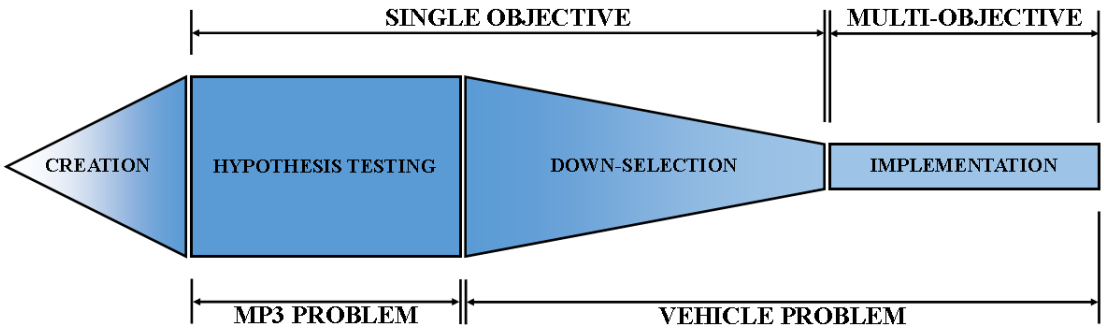

Figure 7.1: Flow Diagram Overview of Presented Research

Within the Down-Selection phase, the crossover operators were then tested on a more robust problem (the single-objective vehicle product line configuration problem) to determine which informed crossover operators offered added computational and/or generational benefits when implemented. It was discovered that the Lowest $n-1$ Share crossover method outperformed

all other informed crossover operators and was ultimately selected for implementation with a problem that was indicative of a true product line optimization being conducted by an engineering firm. The final phase (Implementation) tested the selected informed crossover method on an extremely complicated multi-objective vehicle product line configuration problem to gauge the benefits of this method when applied to a problem more representative of the design decisions faced by firms engineering complex systems.

7.2 Research Question 1

The initial research question posed by this work aimed to determine the advantages of using consumer preference data (and other relevant problem information) to alter crossover techniques applied to product line optimizations.

> **Research Question 1: What are the advantages of using problem-specific information within the crossover operator when optimizing product line configurations?**

The main focus of RQ1 lies in the comparison of the performance of the developed informed crossover operators against the baseline scattered crossover used in this research. An analysis of the data presented within Chapter 4 reveals that objective performance of the developed informed crossover operators matched the baseline scattered crossover. This served to validate the methods, ensuring that no unintended effects were seen due to the implementation of problem-specific data (i.e. a restriction of the design space, convergence upon a local optima, etc.). The generational results from these trials did indicate

improvements in computational efficiency when compared against the baseline scattered crossover. Significant decreases in the number of generations required to reach the reported optimum were discovered through implementation of each informed crossover method. Interestingly, it appeared as if crossover techniques that used data that was more relevant to the objective being optimized seemed to reach an optimum sooner than those that did not (i.e. the Price Sort crossover offered improved computational efficiency when used with a profit maximization objective for the MP3 problem, and the Lowest Share crossover method improved computational efficiency when used with a market share maximization objective). These trends were then tested with the vehicle product line configuration problem.

Test trials for the single-objective maximization with the vehicle product line configuration problem again revealed advantages offered through the implementation of informed crossover operators, though these improvements were lessened due to the complexity of the problem being implemented. Oftentimes improvements in computational efficiency came at the cost of objective performance, implying that some of the informed crossover methods were susceptible to converging on local minima. It is theorized that the problem data used in some of the methods overly reduced the heuristic nature of the standard crossover operator, ultimately at the detriment of solution quality. However, a majority of the test trials again revealed an overall improvement in solution quality and the number of generations required to reach said solution.

The final set of test trials were conducted using the chosen Lowest $n - 1$ Share crossover when applied within a MOGA. The informed crossover operator generally rendered a higher

hypervolume metric, indicating improvements in frontier size and concentration when compared against scattered crossover. These improvements were also noted to expand frontiers at points with higher market share, indicating improved solution quality within these design regions. It is theorized that these improvements are due to the informed crossover methods ability to retain information on the strongest performing product within each line.

7.3 Research Question 2

The focus of RQ2 surrounds comparisons between informed crossover methods, and help to guide the selection of the Lowest $n-1$ Share crossover method as the recommended informed crossover method for product line optimization.

Research Question 2: How does problem formulation impact the performance of crossover operators that use problem-specific data?

As previously noted, the implementation of the informed crossover techniques with a relatively simpler problem (i.e. the MP3 product line problem) revealed that crossover techniques using problem data more closely related to the objective being maximized tended to outperform (both in computational efficiency and algorithm effectiveness) those that did not. However, as the informed crossover methods were applied to more complex problems, the advantages attributed to the Lowest $n-1$ Share crossover method became more pronounced. Many of the informed crossover methods would sacrifice objective maximization for algorithm performance, but the Lowest $n-1$ Share crossover continued to yield improved solution quality in a fewer number of generations than other informed

crossover methods. The Lowest Share crossover method seemed to indicate the potential for improved performance and efficiency with the probabilistic choice rule and random initialization. This is likely due to the similarities between the Lowest Share and Lowest $n-1$ Share crossover methods (in retaining a certain number of strong products).

The final recommendation of the author is to implement the Lowest $n-1$ Share crossover for all product line optimization problems. The crossover method does not require any additional function calls over a standard scattered crossover method, and any increases in computational time added by the method are easily made up for in improvements in solution quality and computational efficiency.

7.4 Future Work

There are numerous avenues that this research can be taken, the most obvious of which is implementation in other problems. Design research often suffers from validation issues due to the complexity of the problems being solved and the uncertainties associated with market predictions and analysis. The easiest means of attempting to overcome this issue is through increased test trials, which can be conducted on various other problems to ensure that the trends noted in this research are repeatable.

Another potential avenue of research is through the development of other informed operators. The research proposed in this thesis leveraged the concept of informed operators from past work on targeted initialization, and it is assumed that further improvements can be realized from the development of informed mutation operators (the logical next step in the genetic

algorithm process). Other potential avenues for informed operator developments include further searches into informed crossover methods and implementation of multiple informed crossover methods (i.e. use an informed operator for 50 generations, then switch over to another informed operator for the remainder of the search).

Future research may also explore the implementation of other sources of problem information within crossover methods. These may include more product attribute related data (such as dependencies between attributes by leveraging DSM data) or the use of schema designed to guarantee combinations of certain products within a product line. Ultimately, the potential for problem data implementation within GA operators is seemingly limitless, as there are a myriad of potential information sources and implementation methods available to a designer for usage due to the heuristic nature of the optimization.

REFERENCES

[1] Michalek, J. J., 2006, "Balancing Marketing and Manufacturing Objectives in Product Line Design," J. Mech. Des., **128**(5), pp. 1196–1204.

[2] Michalek, J. J., Ebbes, P., Adigüzel, F., Feinberg, F. M., and Papalambros, P. Y., 2011, "Enhancing marketing with engineering : Optimal product line design for heterogeneous markets," Int. J. Res. Mark., **28**(1), pp. 1–12.

[3] Sullivan, E., Ferguson, S., and Donndelinger, J., 2011, "Exploring Differences in Preference Heterogeneity Representation and Their Influence in Product Family Design," 37th Design Automation Conference, ASME, pp. 81–92.

[4] Turner, C., Ferguson, S., and Donndelinger, J., 2011, "Exploring Heterogeneity of Customer Preference to Balance Commonality and Market Coverage," ASME 2011 International Design Engineering Technical Conferences & Computers and Information in Engineering Conference, pp. 1–13.

[5] Huber, J., Arora, N., and Johnson, R., 1998, "Capturing Heterogeneity in Consumer Choices," Am. Mark. Assoc.

[6] Kumar, D., Hoyle, C., Chen, W., Wang, N., Levi, G. G., and Koppelman, F. S., 2009, "A hierarchical choice modelling approach for incorporating customer preferences in vehicle package design," Int. J. Prod. Dev., **8**(3), p. 228.

[7] Ferguson, S., Olewnik, A., and Cormier, P., 2004, "Exploring Marketing to Engineering Information Mapping in Mass Customization: A Presentation of Ideas, Challenges and Resulting Questions," ASME Design Engineering Technology Conference.

[8] Pine, B. J., 1993, Mass Customization: the new frontier in business competition, Harvard Business School Press, Boston, MA.

[9] Prahalad, C. K., and Mashelkar, R. A., 2010, "Innovation's Holy Grail," Harv. Bus. Rev., (July-August).

[10] Foster, G., 2014, "Efficient Optimization of Product Lines with Design Prohibitions."

[11] Kwong, C. K., Luo, X. G., and Tang, J. F., 2011, "A Multiobjective Optimization Approach for Product Line Design," IEEE Trans. Evol. Comput., **58**(1), pp. 97–108.

[12] Meyer, M. H., and Lehnerd, A. P., 1997, The Power of Product Platforms: Building Value and Cost Leadership, Free Press, New York.

[13] Guide, Jr., V. D. R., and Li, J., 2010, "The Potential for Cannibalization of New Products Sales by Remanufactured Products ∗," Decis. Sci., **41**(3), pp. 547–572.

[14] Green, P. E., and Rao, V. P., 1971, "Conjoint measurement for quantifying judgemental data," J. Mark. Res., **8**(3), pp. 355–363.

[15] Ben-Akiva, M., and Lerman, S. R., 1985, Discrete Choice Analysis: Theory and Application to Travel Demand, MIT Press, Cambridge, MA.

[16] Moore, W. L., Louviere, J. J., and Verma, R., 1999, "Using Conjoint Analysis to Help Design Product Platforms," J. Prod. Innov. Manag., **16**(1), pp. 27–39.

[17] Train, K. E., 2009, Discrete Choice Methods with Simulation, Cambridge University Press, New York.

[18] Li, H., and Azarm, S., 2000, "Product Design Selection Under Uncertainty and With Competitive Advantage," J. Mech. Des., **122**(4), pp. 411–418.

[19] Hauser, J. R., 2011, "New developments in product-line optimization," Int. J. Res. Mark., **28**(1), pp. 26–27.

[20] Tsafarakis, S., Marinakis, Y., and Matsatsinis, N., 2011, "Particle swarm optimization for optimal product line design," Int. J. Res. Mark., **28**(1), pp. 13–22.

[21] Green, P. E., and Krieger, A. M., 1985, "Models and Heuristics for Product Line Selection," Mark. Sci., **4**(1), pp. 1–19.

[22] Khajavirad, A., Michalek, J. J., and Simpson, T. W., 2009, "An efficient decomposed multiobjective genetic algorithm for solving the joint product platform selection and product family design problem with generalized commonality," Struct. Multidiscip. Optim., **39**(2), pp. 187–201.

[23] Kumar, S., and Chatterjee, A. K., 2013, "A profit maximizing mathematical model for pricing and selecting optimal product line," Comput. Ind. Eng., **64**(2), pp. 545–551.

[24] Nichols, K. B., Venkataramanan, M. A., and Ernstberger, K. W., 2005, "Product Line Selection and Pricing Analysis: Impact of Genetic Relaxations," Math. Comput. Model., **42**(2005), pp. 1397–1410.

[25] Qian, L., and Ben-Arieh, D., 2009, "Joint Pricing and Platform Configuration in Product Family Design With Genetic Algorithm," 14th Design for Manufacturing and the Life Cycle Conference, 6th Symposium on International Design and Design Education, 21st International Conference on Design Theory and Methodology, ASME, pp. 49–58.

[26] Siedlecki, W., and Sklansky, J., 1989, "A note on genetic algorithms for large-scale feature selection," Pattern Recognit. Lett., **10**(5), pp. 335–347.

[27] Luo, L., 2011, "Product Line Design for Consumer Durables: An Integrated Marketing and Engineering Approach," J. Mark. Res., **48**(1), pp. 128–139.

[28] Belloni, A., Freund, R., Selove, M., and Simester, D., 2008, "Optimizing Product Line Designs: Efficient Methods and Comparisons," Manage. Sci., **54**(9), pp. 1544–1552.

[29] Sanderson, S., and Usumeri, M., 1995, "Managing product families : The case of the Sony Walkman," Res. Policy, **24**(1995), pp. 761–782.

[30] May, J. H., Shocker, A. D., and Sudharshan, D., 1983, "A Simulation Comparison of Methods for New Product Location," Mark. Sci., **6**(82-126), pp. 182–201.

[31] Kohli, R., and Krishnamurti, R., 1987, "A Heuristic Approach to Product Design," Manage. Sci., **33**(12), pp. 1523–1533.

[32] Kohli, R., and Sukumar, R., 1990, "Heuristics for Product-Line Design Using Conjoint Analysis," Manage. Sci., **36**(12), pp. 1464–1478.

[33] Nair, Suresh K.Thakur, L. S. K.-W. W., 1995, “Near Optimal Solutions for Product Line Design and Selection: Beam Search Heuristics.,” Manage. Sci., **41**(5), pp. 767–785.

[34] Thakur, L. S., Nair, S. K., Wen, K.-W., and Tarasewich, P., 2000, “A new model and solution method for product line design with pricing,” J. Oper. Res. Soc., **51**(1), pp. 90–101.

[35] Sivanandam, S. N., and Deepa, S. N., 2007, Introduction to Genetic Algorithms, Springer, Berlin, NY.

[36] Green, P. E., Wind, Y., and Jain, A. K., 1972, “Consumer Menu Preference: An Application of Additive Conjoint Measurement,” 3rd Annual Conference of the Association for Consumer Research, pp. 304–315.

[37] Shiau, C.-S. N., and Michalek, J. J., 2009, “Optimal Product Design Under Price Competition,” J. Mech. Des., **131**(7), pp. 1–10.

[38] Giesen, J., Mueller, K., Taneva, B., and Zolliker, P., 2011, Choice-Based Conjoint Analysis: Classification vs. Discrete Choice Models, Preference Learning, Heidelberg.

[39] Raghavarao, D., Chitturi, P., and Wiley, J. B., 2011, Choice-based conjoint analysis: models and designs, Chapman & Hall/CRC., Boca Raton.

[40] Inc., S. S., 2013, “The CBC System for Choice-Based Conjoint Analysis,” Sawtooth Softw. Tech. Pap. Ser., pp. 1–27.

[41] Sawtooth Software Inc., 2013, “SSI Web.”

[42] Qualtrics, 2014, “Research Suite.”

[43] Sawtooth Software, I., 2009, "CBC Hierarchical Bayes Module."

[44] Thurston, D. L., 1991, "A Formal Method for Subjective Design Evaluation with Multiple Attributes," Res. Eng. Des., **3**(2), pp. 105–122.

[45] Bradley, R. A., 1954, "Rank Analysis of Incomplete Block Designs : II . Additional Tables for the Method of Paired Comparisons," Biometrika, **41**(3/4), pp. 502–537.

[46] Orme, B. K., 2010, "Market Simulators for Conjoint Analysis," Getting Started with Conjoint Analysis: Strategies for Product Design and Pricing Research, pp. 89–103.

[47] Cook, H. E., and DeVor, R. E., 1991, "On Competitive Manufacturing Enterprises I: The S-Model and the Theory of Quality," Manuf. Rev., **4**(2), pp. 96–105.

[48] Hazelrigg, G. A., 1998, "A Framework for Decision-Based Engineering Design," J. Mech. Des., **120**(4), pp. 653–658.

[49] Michalek, J. J., Feinberg, F. M., and Papalambros, P. Y., 2005, "Linking Marketing and Engineering Product Design Decisions via Analytical Target Cascading," J. Prod. Innov. Manag., **22**(1), pp. 42–62.

[50] Wassenaar, H., Chen, W., Chen, J., and Sudjianto, A., 2004, "An integrated latent variable choice modeling approach for enhancing product demand modeling," ASME Design Engineering Technology Conference.

[51] Jocelyn, X., Camm, J. D., Curry, D. J., Wane, X. T., and Curry, D. J., 2009, "A Branch-and-Price Approach to the Share-of-Choice Product Line Design Problem," Manage. Sci., **55**(10), pp. 1718–1728.

[52] Dobson, G., Kalish, S., and Graduate, W. E. S., 1993, "Heuristics for Pricing and Positioning a Product-Line Using Conjoint and Cost Data," Manage. Sci., **39**(2), pp. 160–175.

[53] Fujita, K., 2002, "Product variety optimization under modular architecture," Comput. Des., **34**(2002), pp. 953–965.

[54] Turner, C., Foster, G., Ferguson, S., and Donndelinger, J., 2012, "Targeted Initial Populations for Multiobjective Product Line Optimization," AIAA Multidisciplinary Analysis and Optimization Conference, Indianapolis, IN.

[55] Goldberg, D. E., 1989, Genetic Algorithms in Search, Optimization, and Machine Learning, Addison-Wesley Professional.

[56] Simpson, T. W., 2004, "Product platform design and customization: Status and promise," Artif. Intell. Eng. Des. Anal. Manuf., **18**(1), pp. 3–20.

[57] Deb, K., Member, A., Pratap, A., Agarwal, S., and Meyarivan, T., 2002, "A Fast and Elitist Multiobjective Genetic Algorithm: NSGA-II," IEEE Trans. Evol. Comput., **6**(2), pp. 182–197.

[58] Woodruff, M. J., Hadka, D. M., Reed, P. M., and Simpson, T. W., 2012, "Auto-Adaptive Search Capabilities of the New Borg MOEA: A Detailed Comparison on Alternative Product Family Design Problem Formulations," 12th AIAA Aviation Technology, Integration and Operations (ATIO) Conference and 14th AIAA/ISSM.

[59] The MathWorks, I., 2008, "MATLAB - The Language of Technical Computing" [Online]. Available: http://www.mathworks.com/products/matlab/.

[60] Balakrishnan, P. V. S., Gupta, R., and Jacob, V. S., 2006, "An investigation of mating and population maintenance strategies in hybrid genetic heuristics for product line designs," Comput. Oper. Res., **33**(2006), pp. 639–659.

[61] D'Souza, B., and Simpson, T. W., 2003, "A genetic algorithm based method for product family design optimization," Eng. Optim., **35**(1), pp. 1–18.

[62] Foster, G., Turner, C., Ferguson, S., Donndelinger, J., Foster, G., Turner, C., Ferguson, S., and Donndelinger, J., 2014, "Creating targeted initial populations for genetic product searches in heterogeneous markets," Eng. Optim., **46**(12), pp. 1729–1747.

[63] Foster, G., and Ferguson, S., 2013, "Enhanced Targeted Populations for Multiobjective Product Line Optii," ASME International Design Engineering Technical Conference and Computers and Informatoin in Engineering Conference, pp. 1–13.

[64] Zhou, C., Zheng, J., and Li, W., 2008, "An improved heuristic crossover operator for TSP," Fourth International Conference on Natural Computation, pp. 541–545.

[65] Vahdati, G., and Yaghoubi, M., 2009, "A New Approach to Solve Traveling Salesman Problem Using Genetic Algorithm Based on Heuristic Crossover and Mutation Operator," International Conference of Soft Computing and Pattern Recognition, pp. 112–116.

[66] Ho, S.-Y., and Lee, K.-Z., 2003, "Genetic algorithm with intelligent crossover for colour quantization," Imaging Sci. J., **51**(3), pp. 151–162.

[67] Garcia-Martinez, C., Lozano, M., Herrerea, F., Molina, D., and Sanchez, A. M., 2008, "Global and local real-coded genetic algorithms based on parent-centric crossover operators," Eur. J. Oper. Res., **185**(3), pp. 1088–1113.

[68] Nazif, H., and Soon, L., 2012, "Optimised crossover genetic algorithm for capacitated vehicle routing problem," Appl. Math. Model., **36**(5), pp. 2110–2117.

[69] Foster, G., and Ferguson, S., 2013, "Exploring the Effectiveness of Using Graveyard Data When Generating Design Alternatives," J. Comput. Inf. Sci. Eng., **13**(4), pp. 16899–16910.

[70] Stump, G., Lego, S., Yukish, M., Simpson, T. W., and Donndelinger, J. A., 2009, "Visual Steering Commands for Trade Space Exploration: User-Guided Sampling with Example," J. Comput. Inf. Sci. Eng., **9**(4), pp. 1–10.

[71] Besharati, B., Luo, L., Azarm, S., and Kannan, P. K., 2004, "An Integrated Robust Design and Marketing Approach for Product Design Selection Process," ASME 2004 International Design Engineering Technical Conferences, Salt Lake City, UT.

[72] Besharati, B., Luo, L., Azarm, S., and Kannan, P. K., 2006, "Multi-Objective Single Product Robust Optimization: An Integrated Design and Marketing Approach," J. Mech. Des., **128**(4), pp. 884–892.

[73] Khajavirad, A., Michalek, J. J., and Simpson, T. W., 2009, "An Efficient Decomposed Multiobjective Genetic Algorithm for Solving the Joint Product Platform Selection and Product Family Design problem with Generalized Commonality," Struct. Multidiscip. Optim., **39**(2), pp. 187–201.

[74] Simpson, T. W., Maier, J. R. A., and Mistree, F., 2001, "Product platform design: method and application," Res. Eng. Des., **13**(1), pp. 2–22.

[75] Luo, L., Kannan, P. K., Besharati, B., and Azarm, S., 2005, "Design of Robust New Products Under Variability: Marketing Meets Design," J. Prod. Innov. Manag., **22**(2), pp. 177–192.

[76] Pareto, V., 1971, Manual of Political Economy, Augusutus M, New York.

[77] Wu, J., and Azarm, S., 2000, "Metrics for Quality Assessment of a Multiobjective Design Optimization Solution Set," J. Mech. Des., **123**(1), pp. 18–25.

[78] Kim, I. Y., and Weck, O. L. de, 2005, "Adaptive weighted-sum method for bi-objective optimization: Pareto front generation," Struct. Multidiscip. Optim., **29**(2), pp. 149–158.

[79] While, L., Member, S., Hingston, P., Barone, L., and Huband, S., 2006, "A Faster Algorithm for Calculating Hypervolume," IEEE Trans. Evol. Comput., **10**(1), pp. 29–38.

APPENDICES

Appendix A: Experimental Design Results for Single Objective Optimization of Vehicle Product Line Configuration Problem

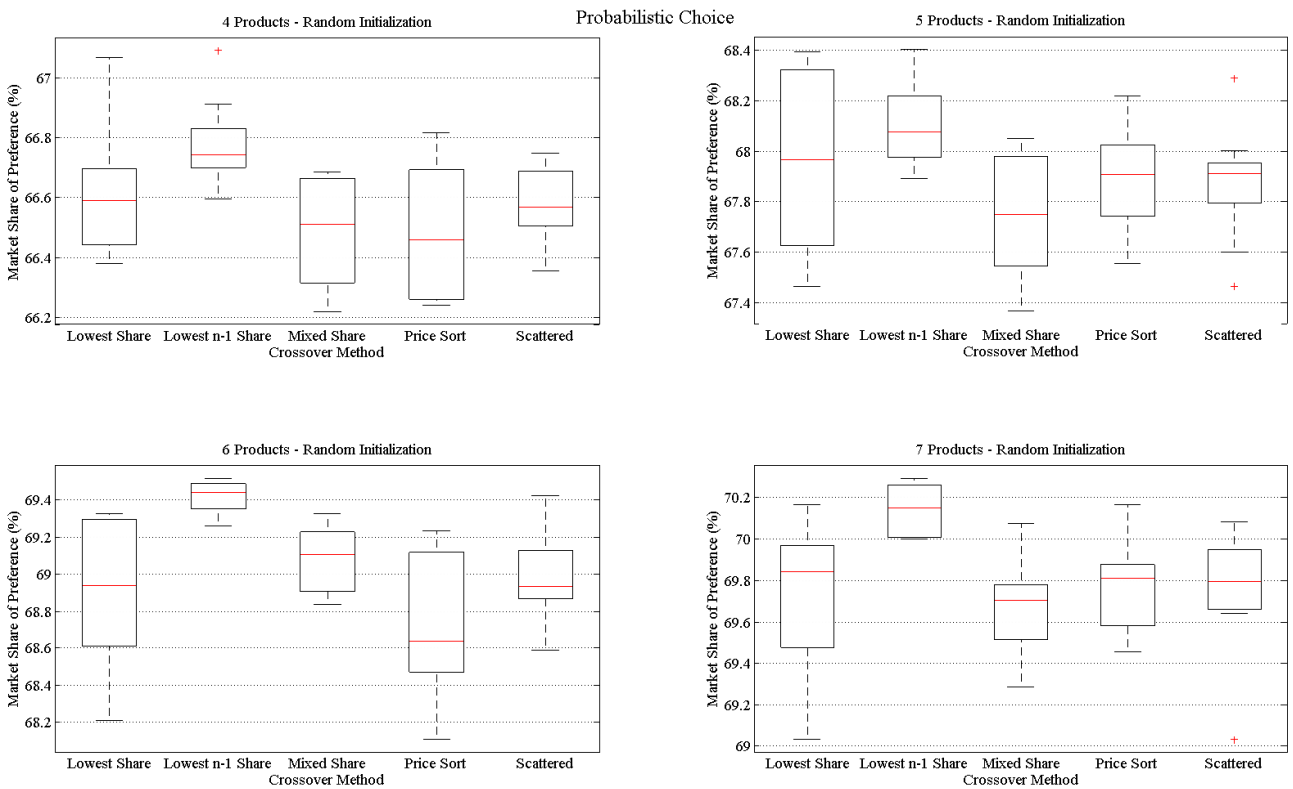

Figure A.1: Objective Results for Vehicle Case Study Using a Probabilistic Choice Rule, Random Initialization, and Maximizing Market Share of Preference

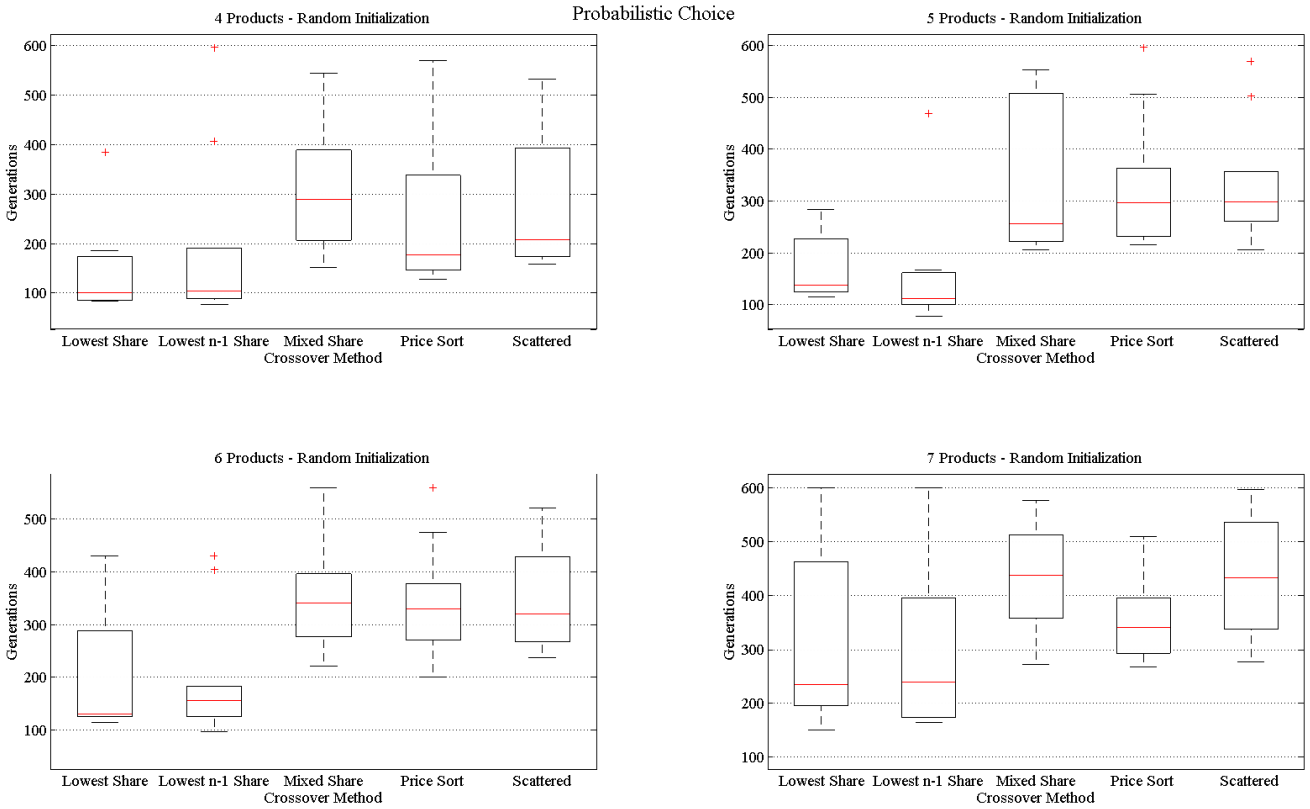

Figure A.2: Generational Results for Vehicle Case Study Using a Probabilistic Choice Rule, Random Initialization, and Maximizing Market Share of Preference

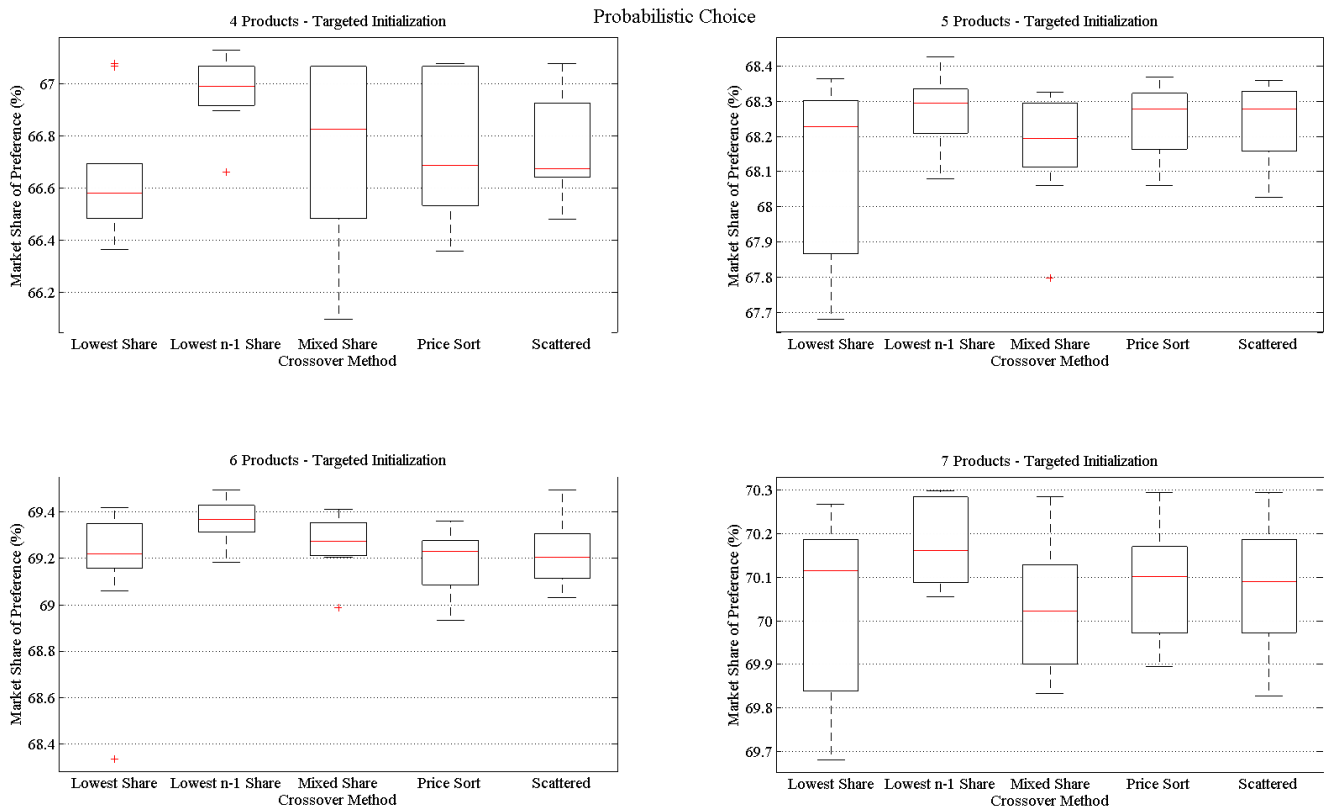

Figure A.3: Objective Results for Vehicle Case Study Using a Probabilistic Choice Rule, Targeted Initialization, and Maximizing Market Share of Preference

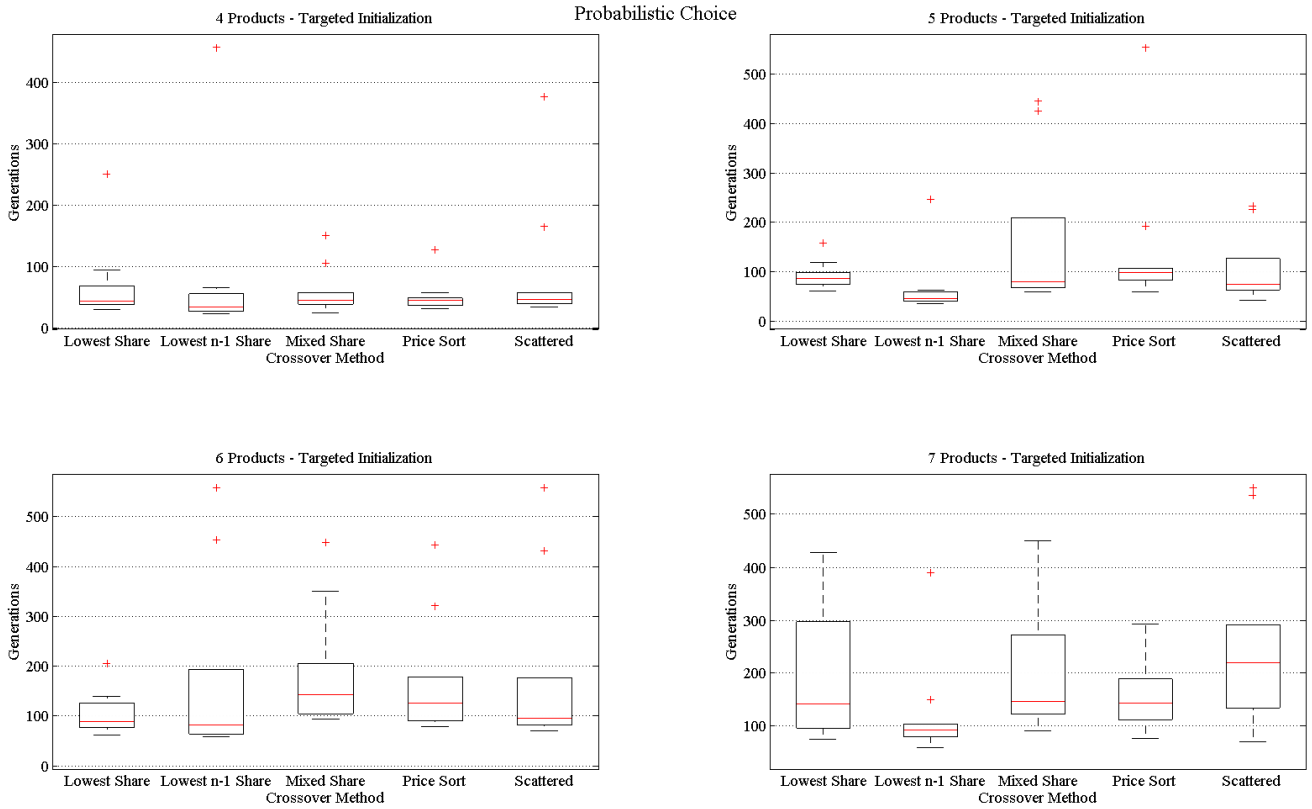

Figure A.4: Generational Results for Vehicle Case Study Using a Probabilistic Choice Rule, Targeted Initialization, and Maximizing Market Share of Preference

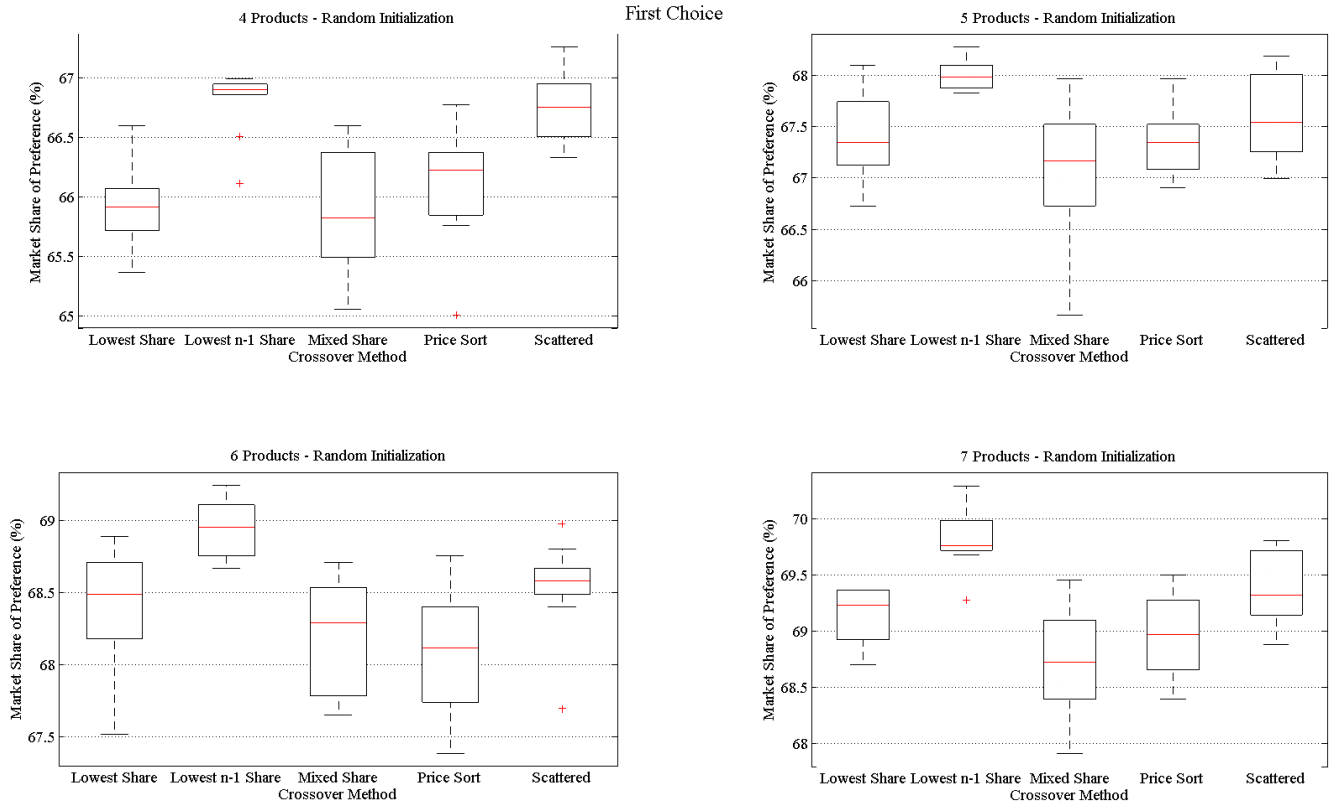

Figure A.5: Objective Results for Vehicle Case Study Using a First Choice Rule, Random Initialization, and Maximizing Market Share of Preference

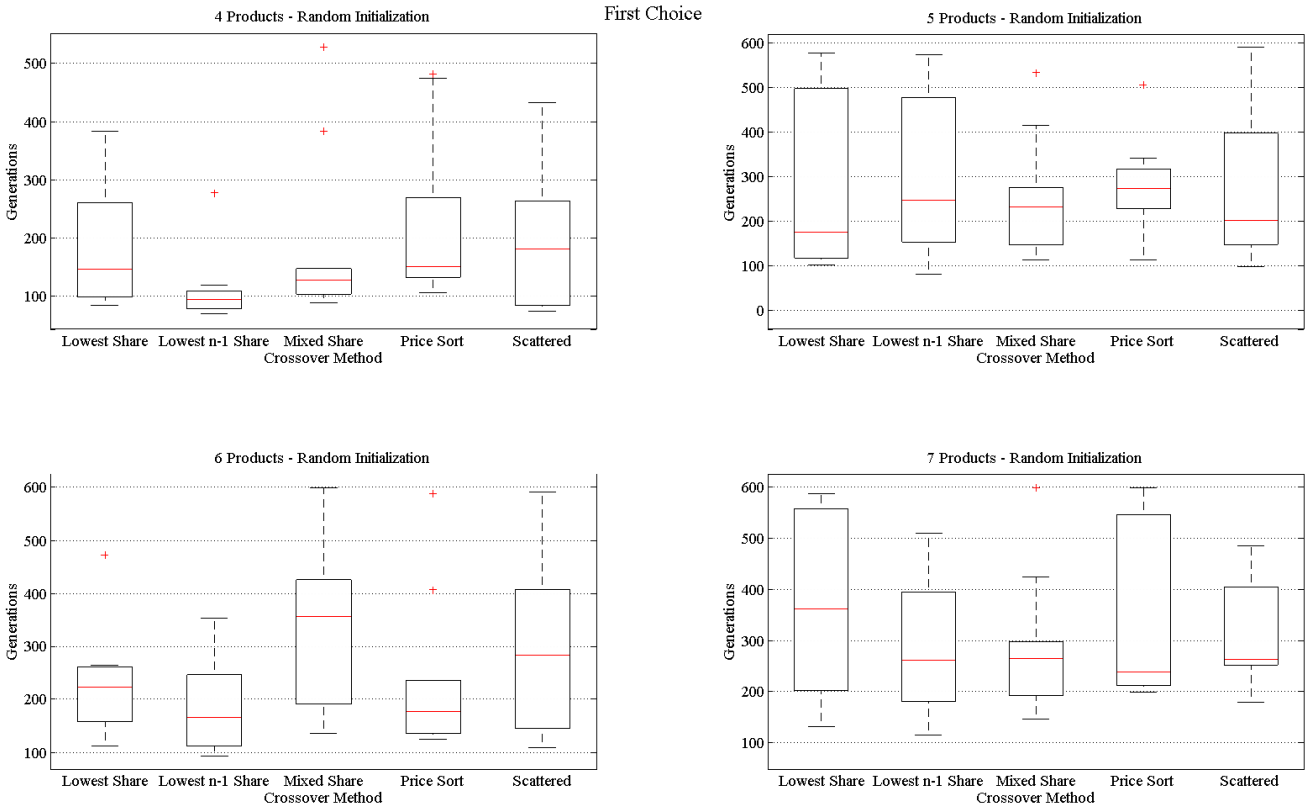

Figure A.6: Generational Results for Vehicle Case Study Using a First Choice Rule, Random Initialization, and Maximizing Market Share of Preference

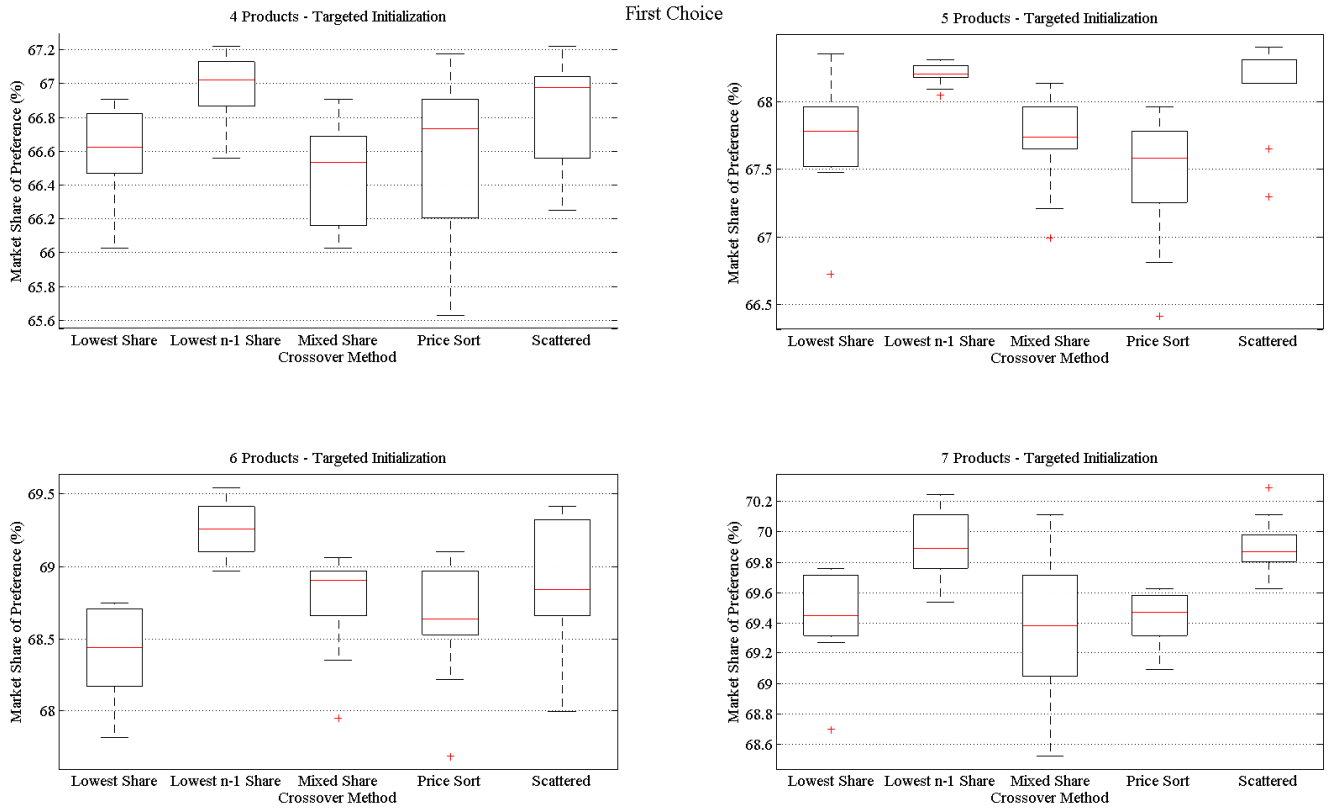

Figure A.7: Objective Results for Vehicle Case Study Using a First Choice Rule, Targeted Initialization, and Maximizing Market Share of Preference

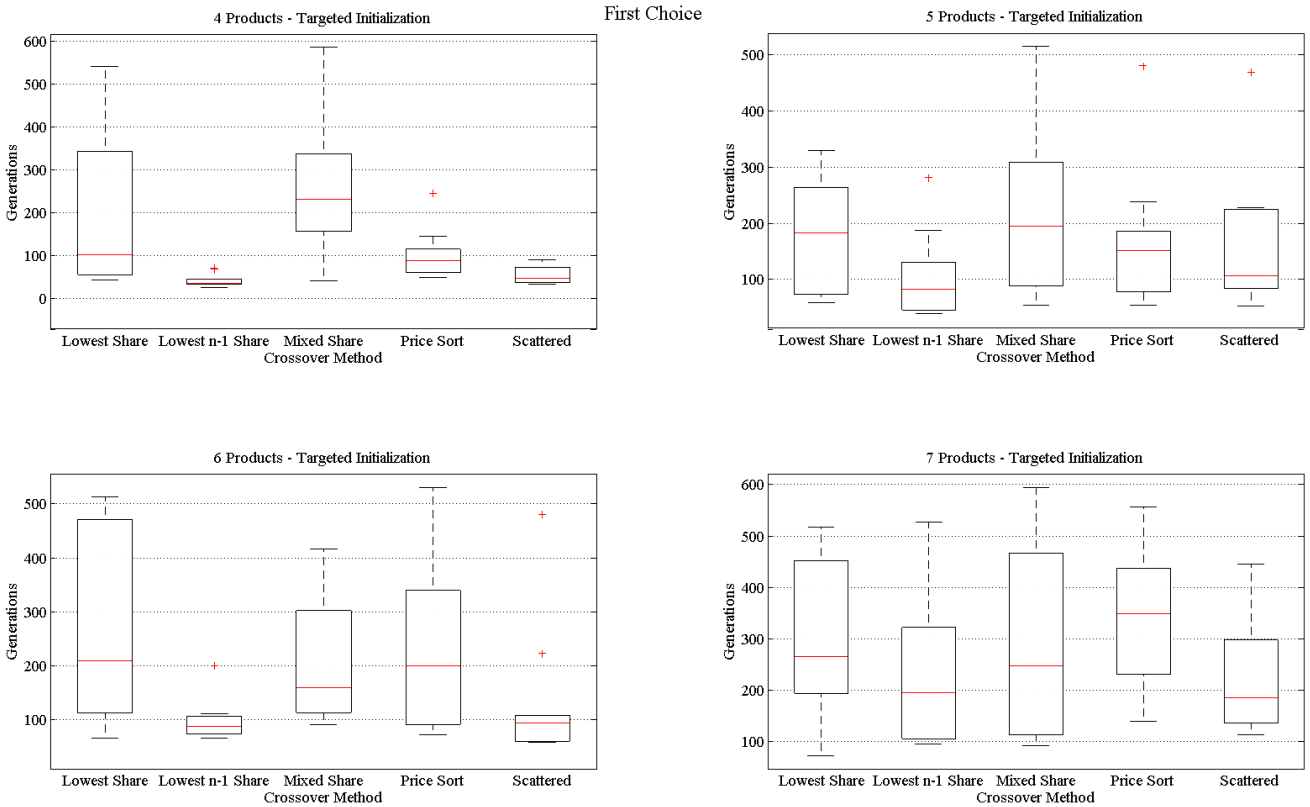

Figure A.8: Generational Results for Vehicle Case Study Using a First Choice Rule, Targeted Initialization, and Maximizing Market Share of Preference

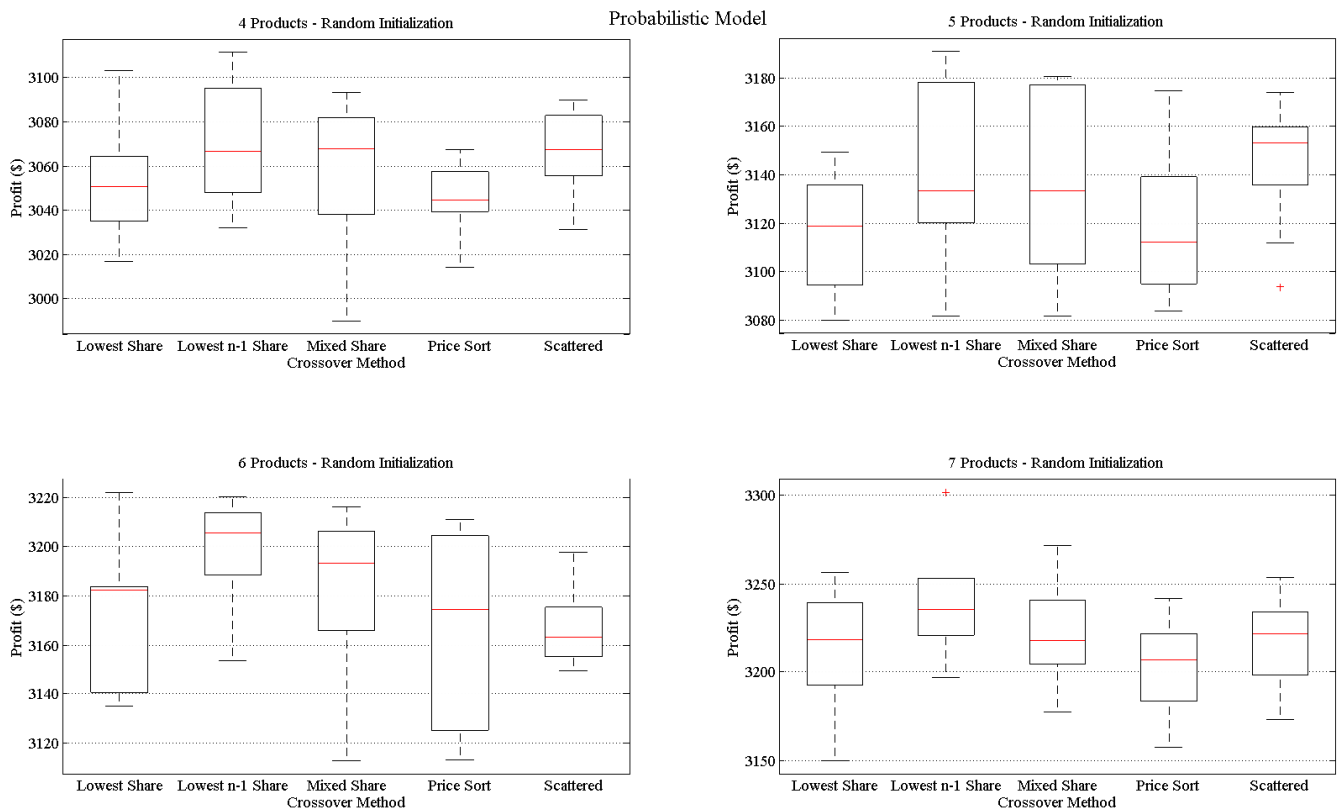

Figure A.9: Objective Results for Vehicle Case Study Using a Probabilistic Choice Rule, Random Initialization, and Maximizing Per Capita Contribution Margin

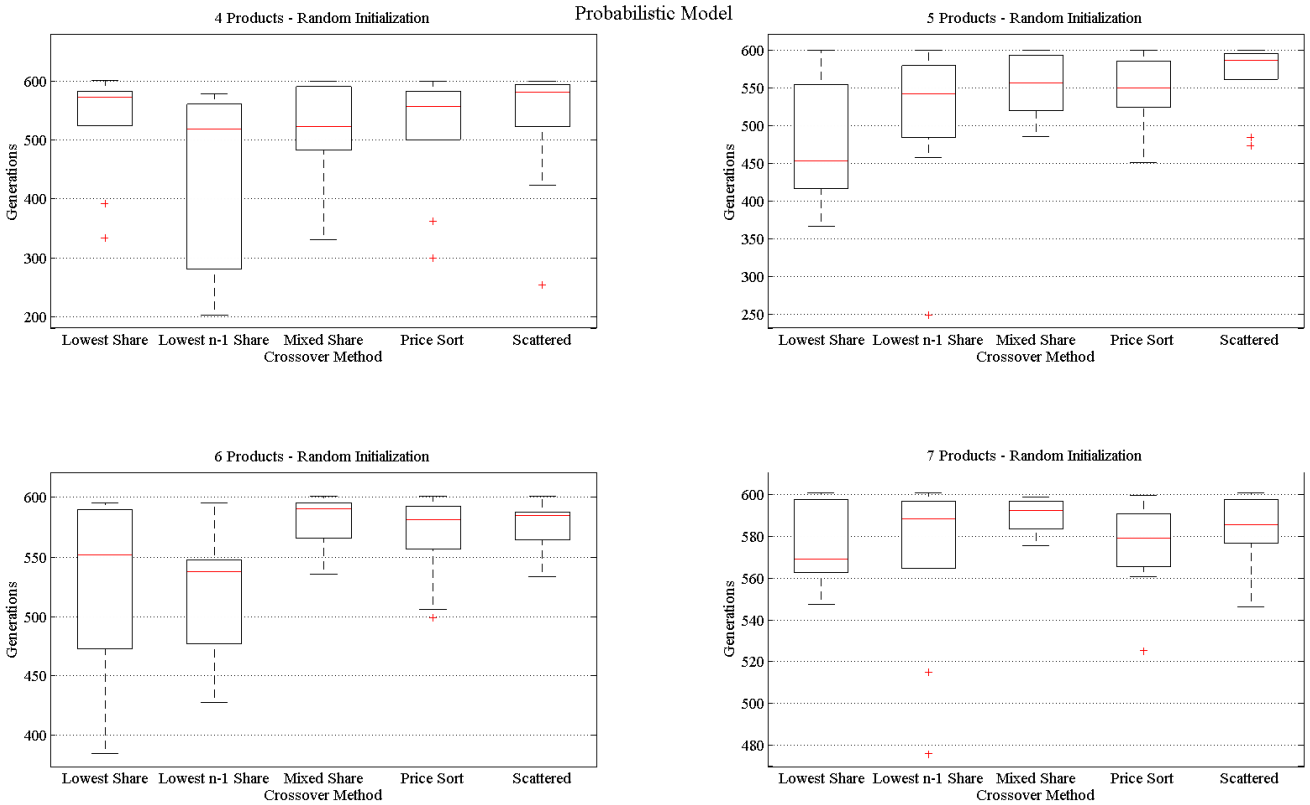

Figure A.10: Generational Results for Vehicle Case Study Using a Probabilistic Choice Rule, Random Initialization, and Maximizing Per Capita Contribution Margin

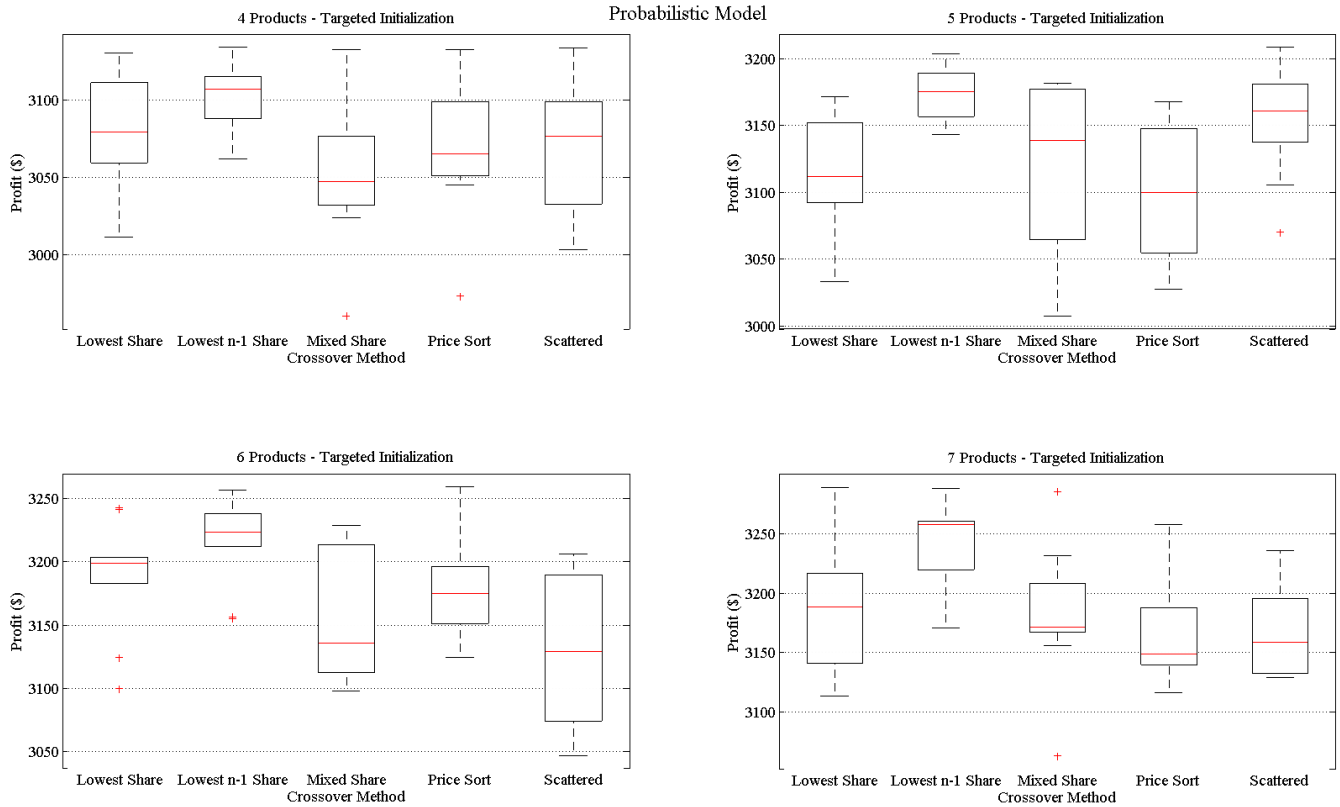

Figure A.11: Objective Results for Vehicle Case Study Using a Probabilistic Choice Rule, Targeted Initialization, and Maximizing Per Capita Contribution Margin

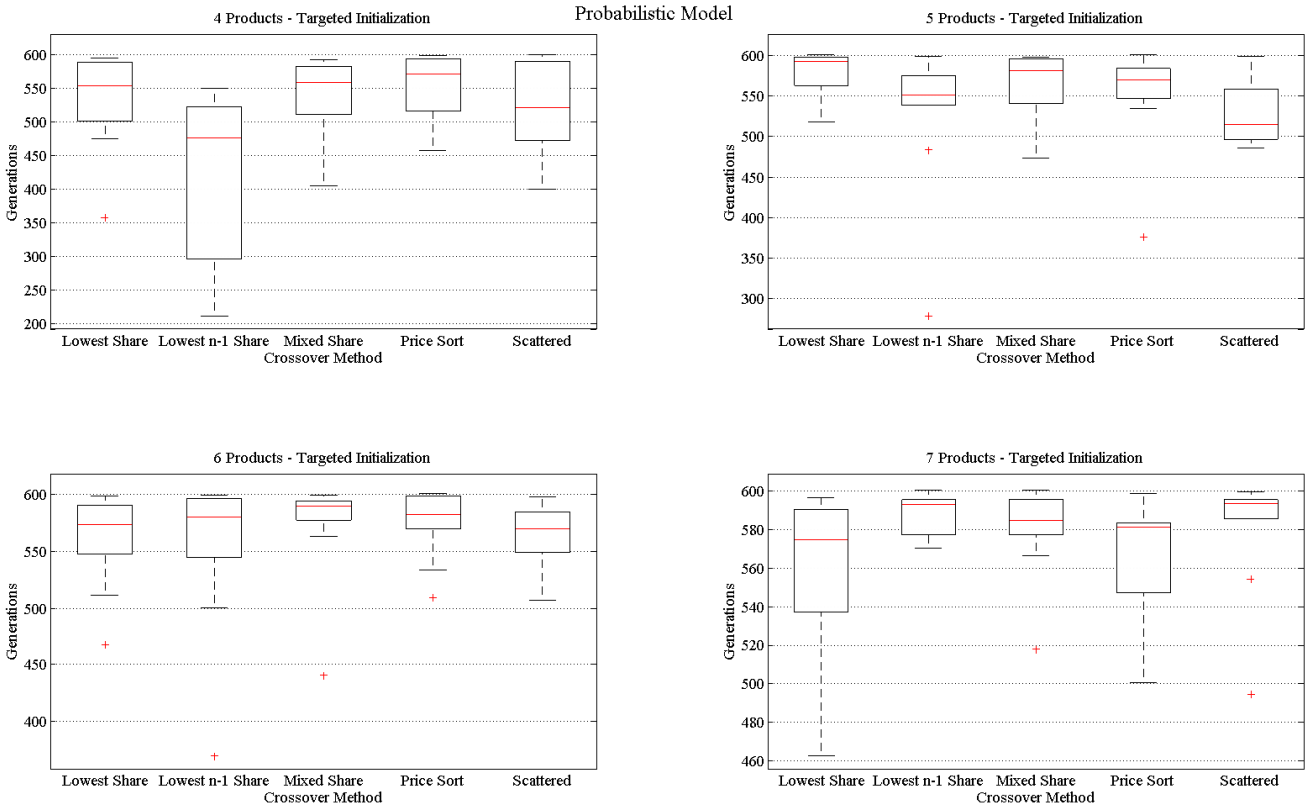

Figure A.12: Generational Results for Vehicle Case Study Using a Probabilistic Choice Rule, Targeted Initialization, and Maximizing Per Capita Contribution Margin

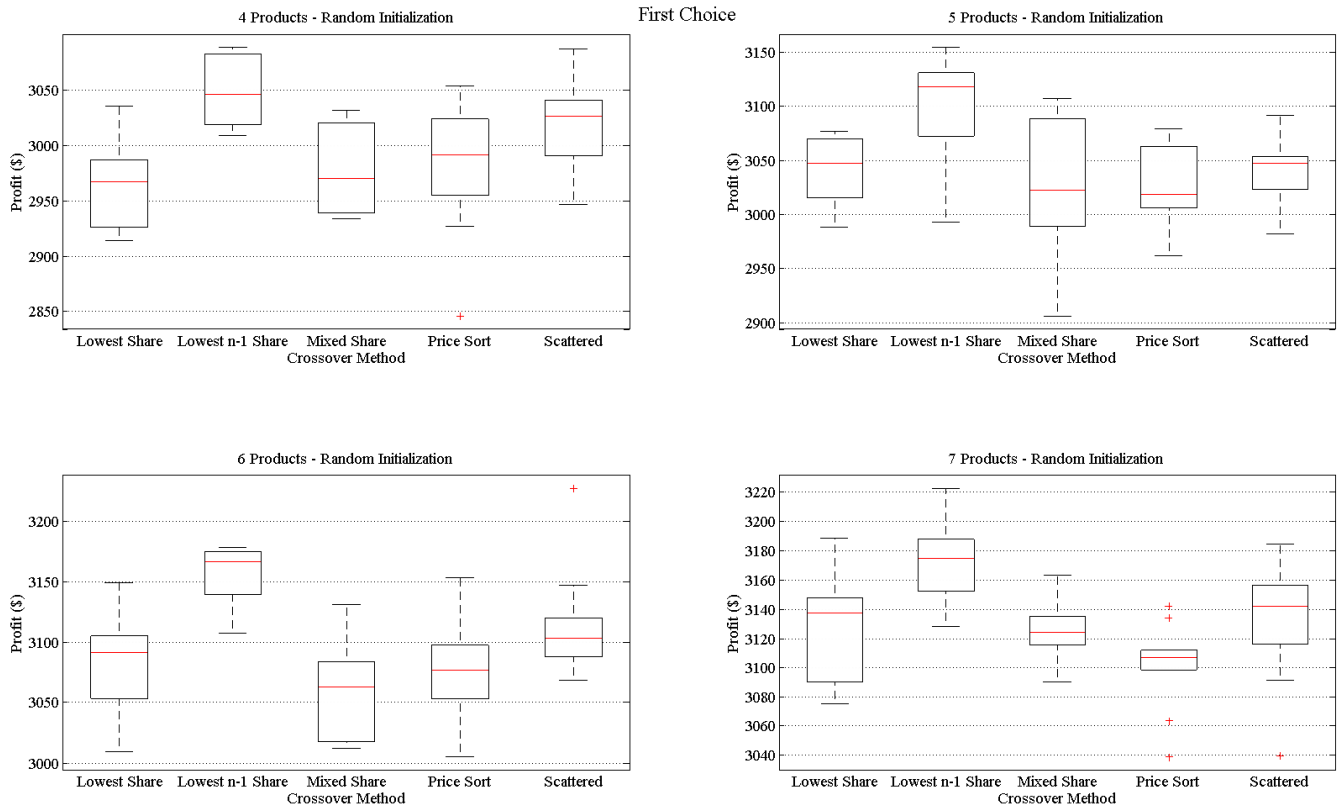

Figure A.13: Objective Results for Vehicle Case Study Using a First Choice Rule, Random Initialization, and Maximizing Per Capita Contribution Margin

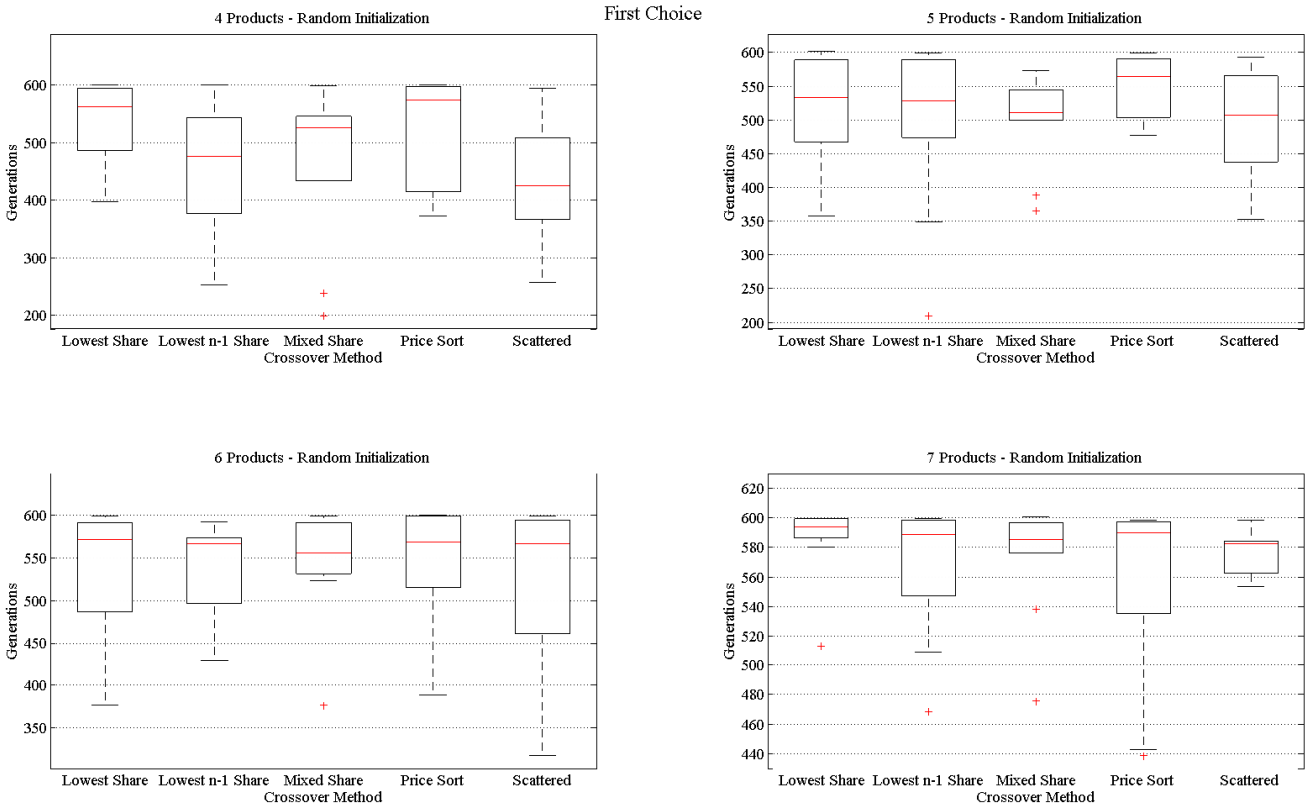

Figure A.14: Generational Results for Vehicle Case Study Using a First Choice Rule, Random Initialization, and Maximizing Per Capita Contribution Margin

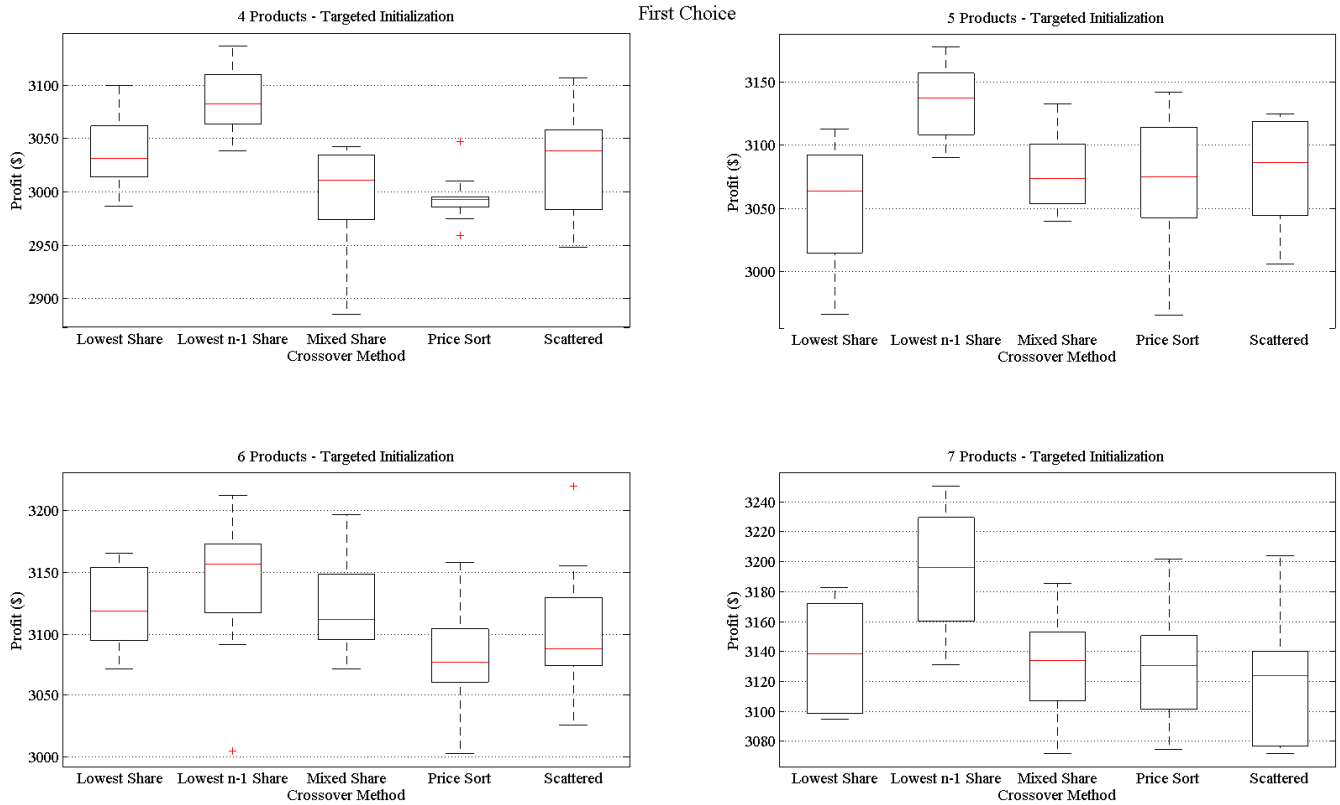

Figure A.15: Objective Results for Vehicle Case Study Using a First Choice Rule, Targeted Initialization, and Maximizing Per Capita Contribution Margin

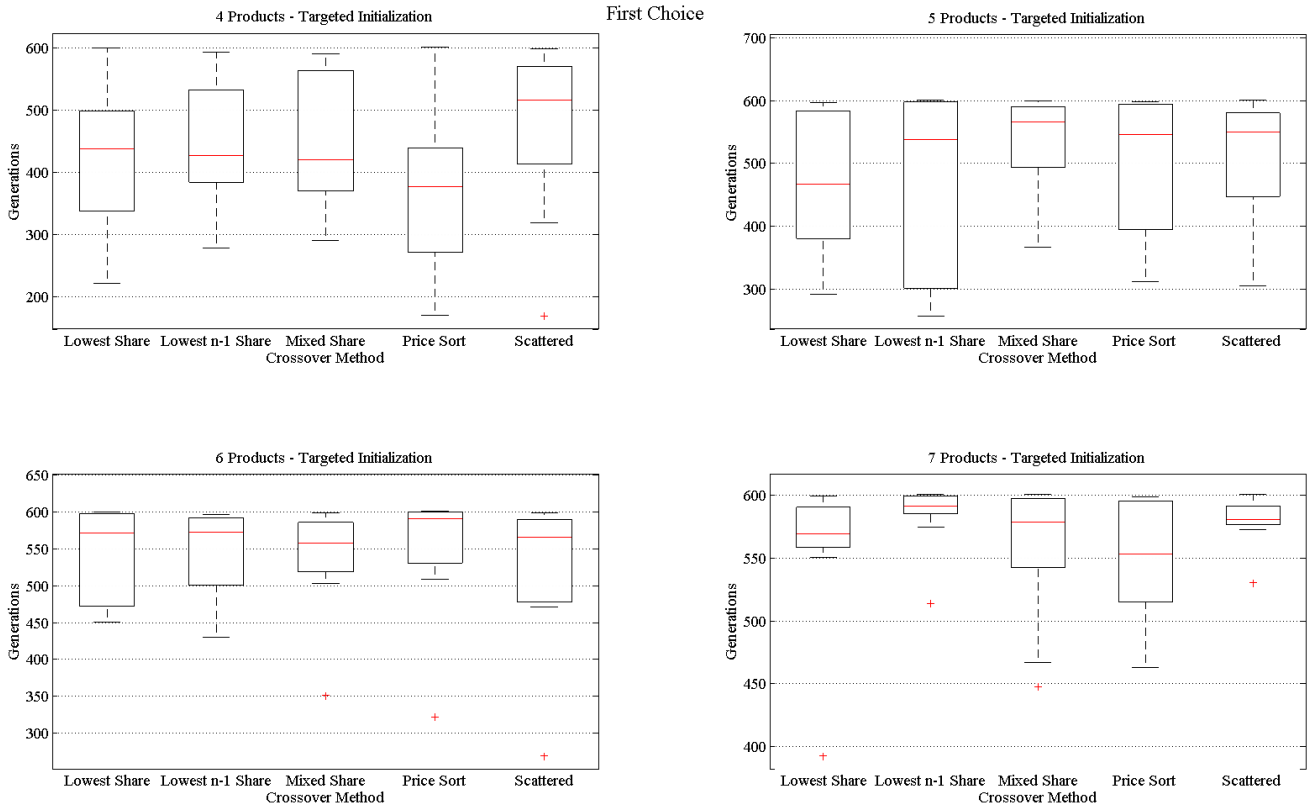

Figure A.16: Generational Results for Vehicle Case Study Using a First Choice Rule, Targeted Initialization, and Maximizing Per Capita Contribution Margin

Appendix B: Pareto Frontiers for Multi-Objective Optimization of Vehicle Product Line Configuration Problem

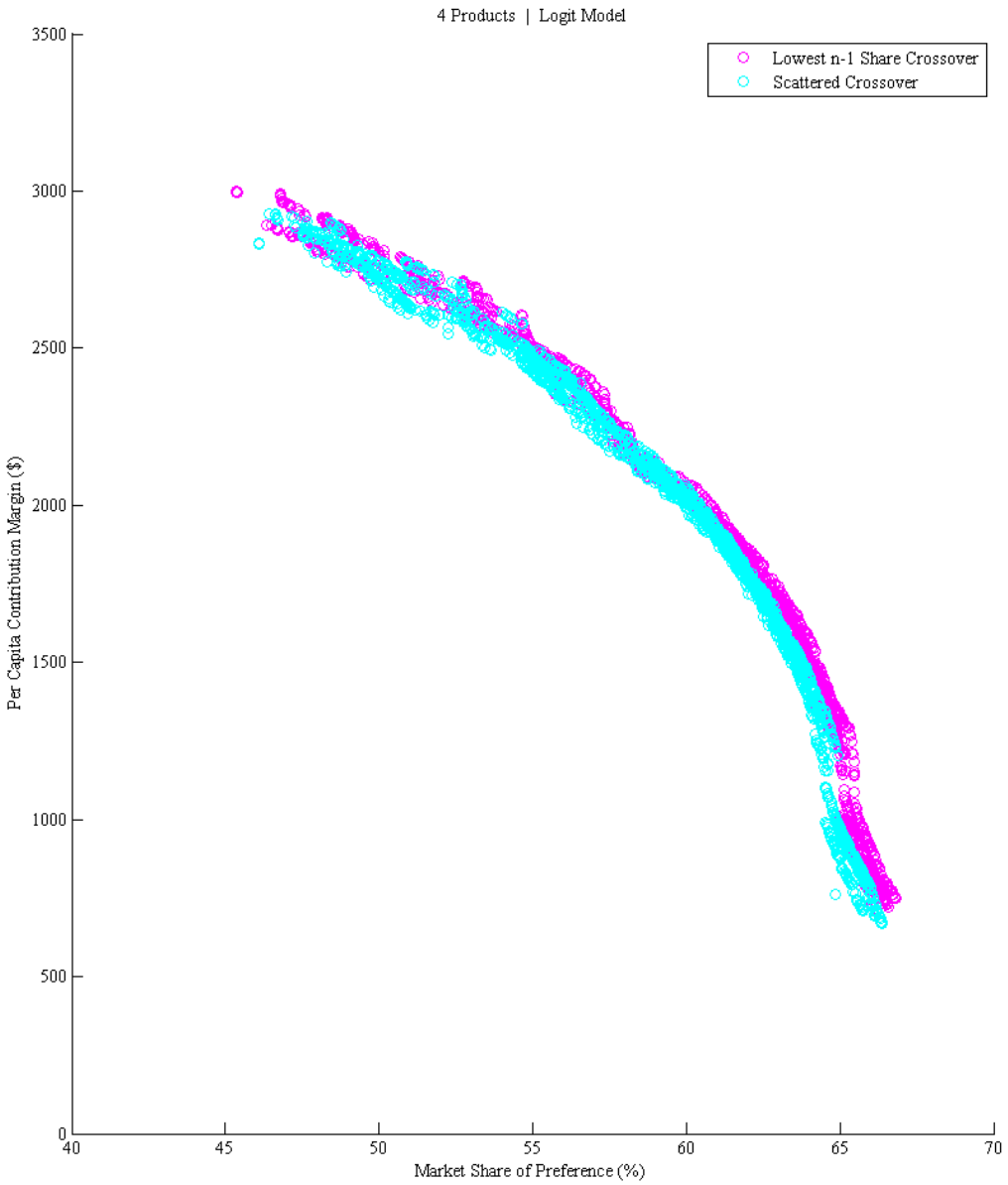

Figure B.1: Pareto Frontiers for Multi-Objective Optimization of Vehicle Product Line Configuration Using Probabilistic Choice Rule, Random Initialization, and 4 Products

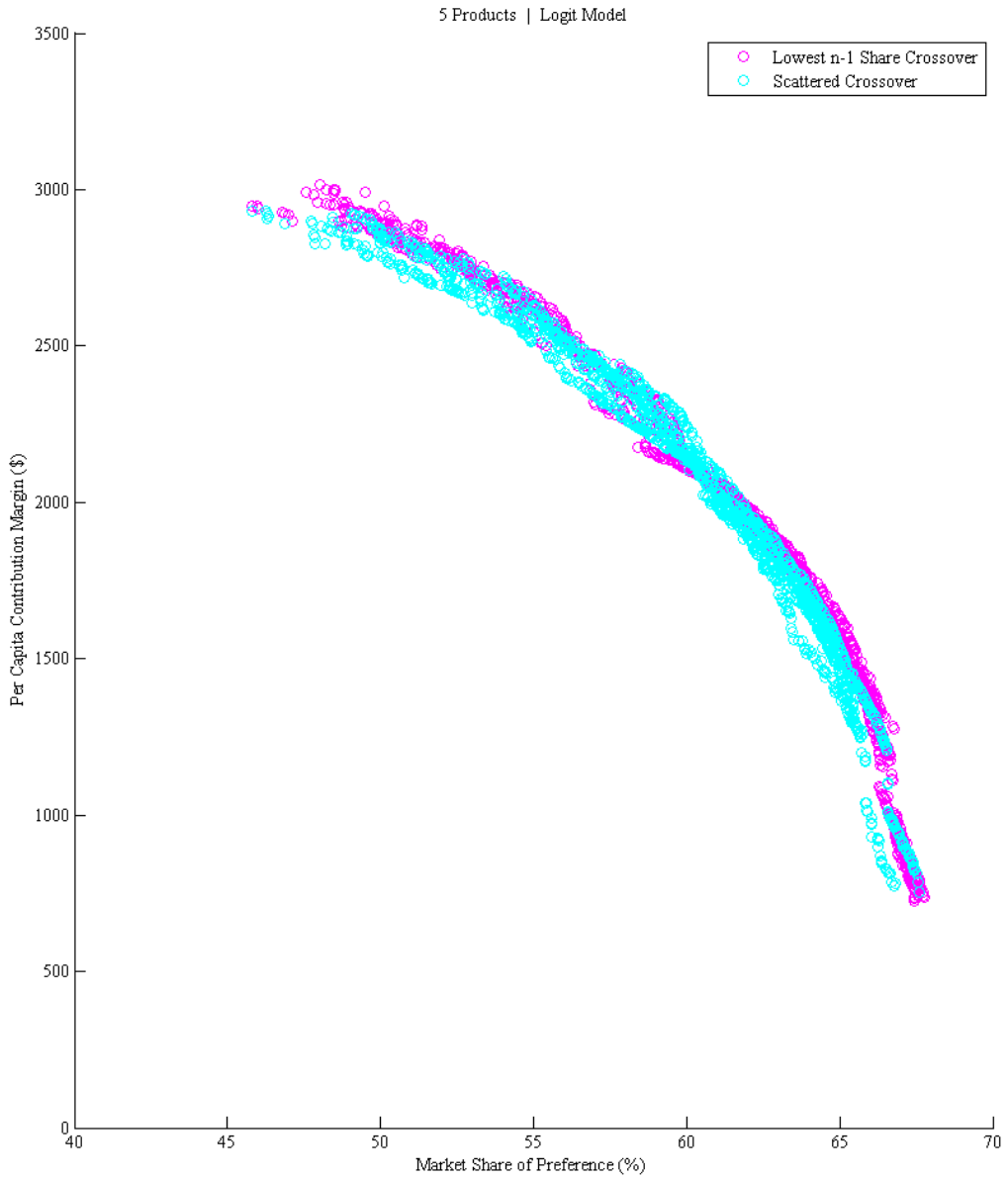

Figure B2: Pareto Frontiers for Multi-Objective Optimization of Vehicle Product Line Configuration Using Probabilistic Choice Rule, Random Initialization, and 5 Products

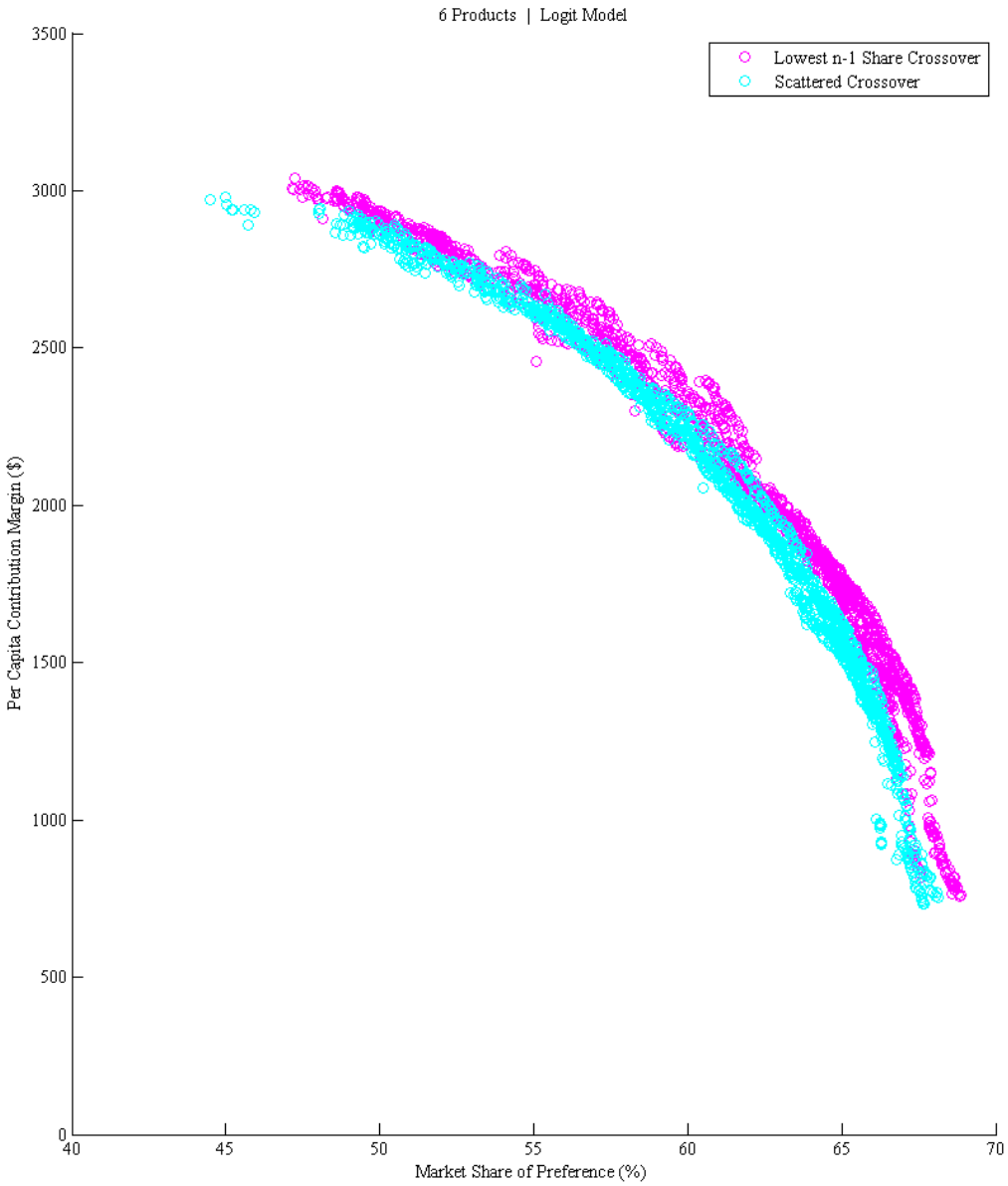

Figure B.3: Pareto Frontiers for Multi-Objective Optimization of Vehicle Product Line Configuration Using Probabilistic Choice Rule, Random Initialization, and 6 Products

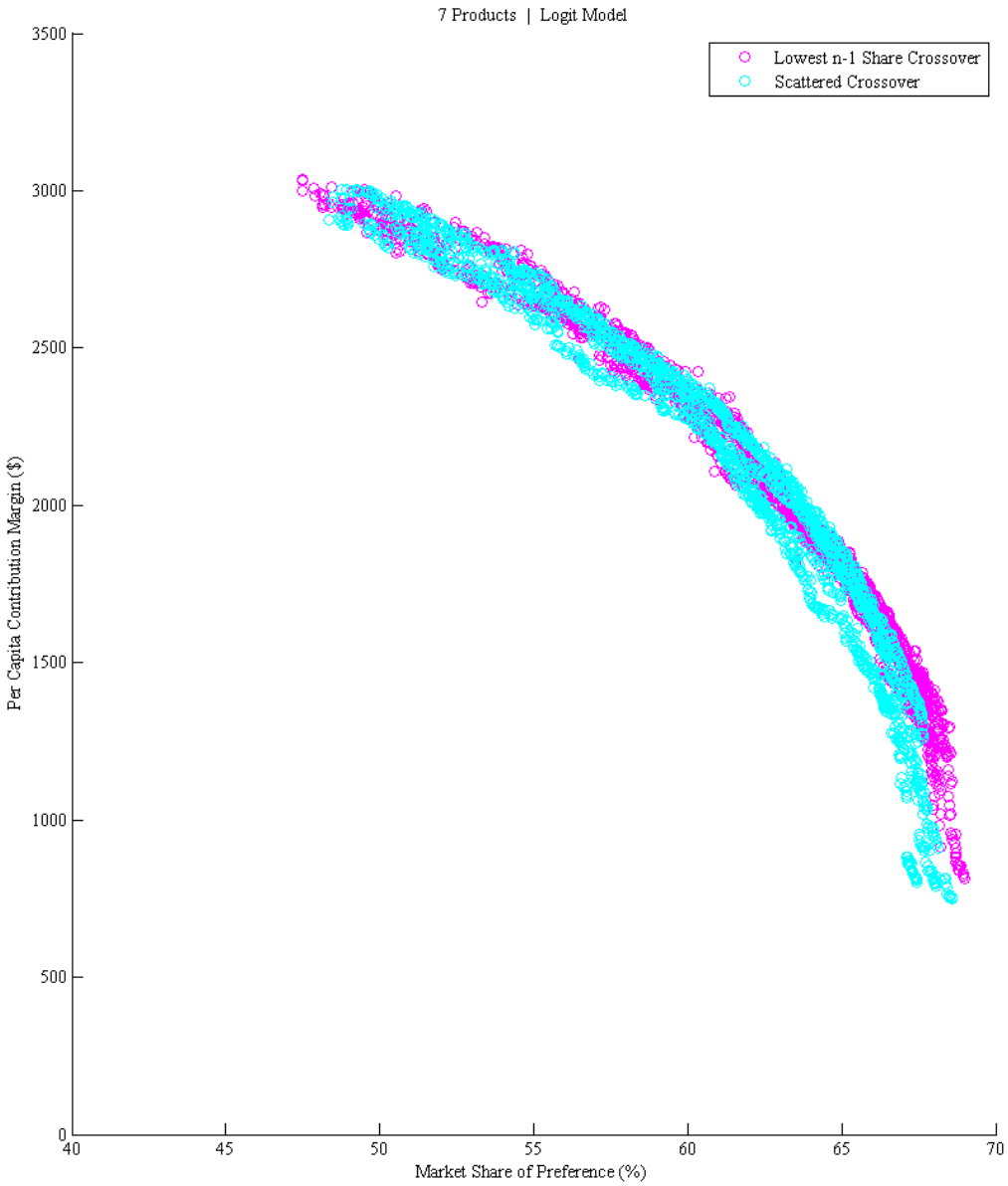

Figure B.4: Pareto Frontiers for Multi-Objective Optimization of Vehicle Product Line Configuration Using Probabilistic Choice Rule, Random Initialization, and 7 Products

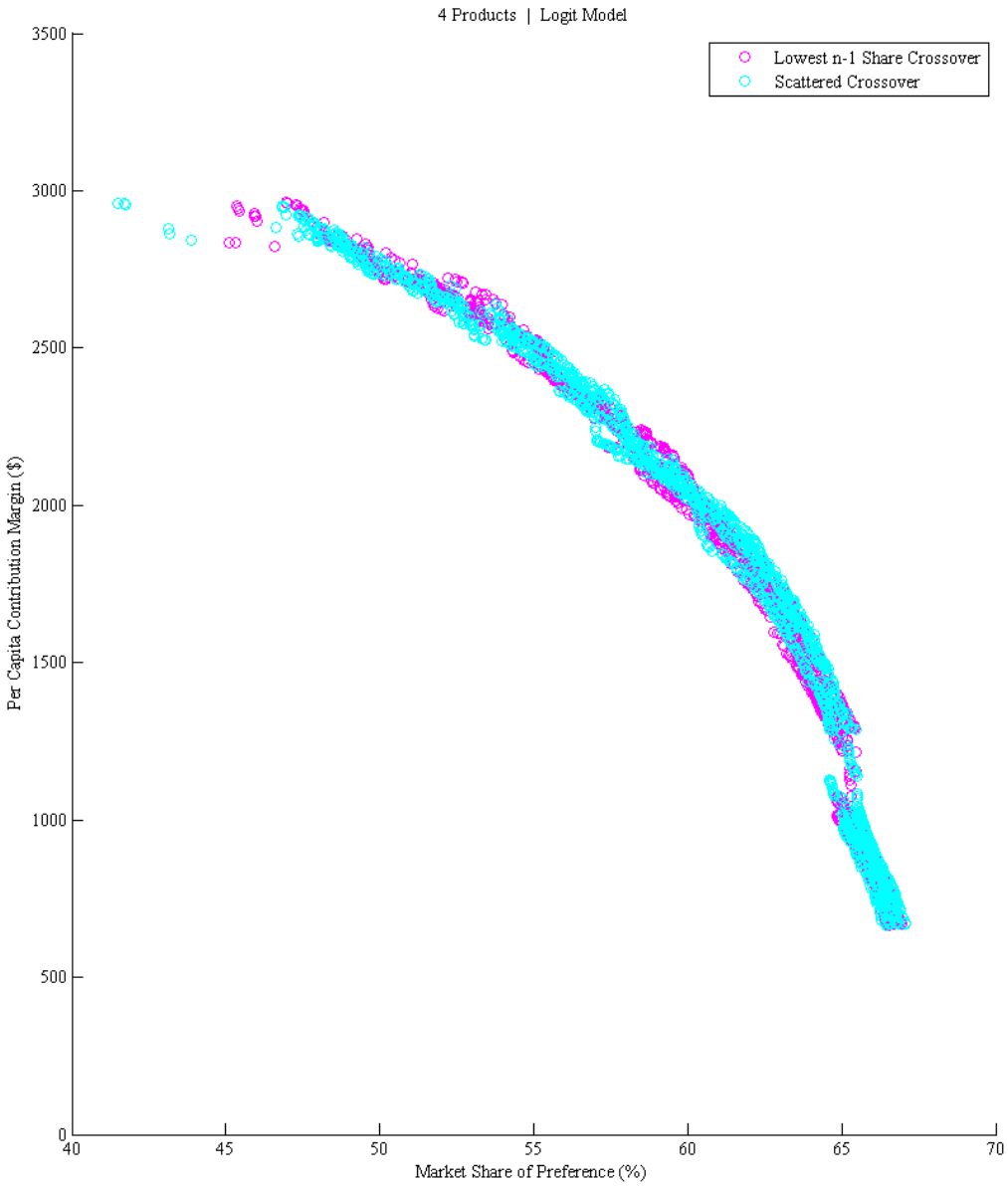

Figure B.5: Pareto Frontiers for Multi-Objective Optimization of Vehicle Product Line Configuration Using Probabilistic Choice Rule, Targeted Initialization, and 4 Products

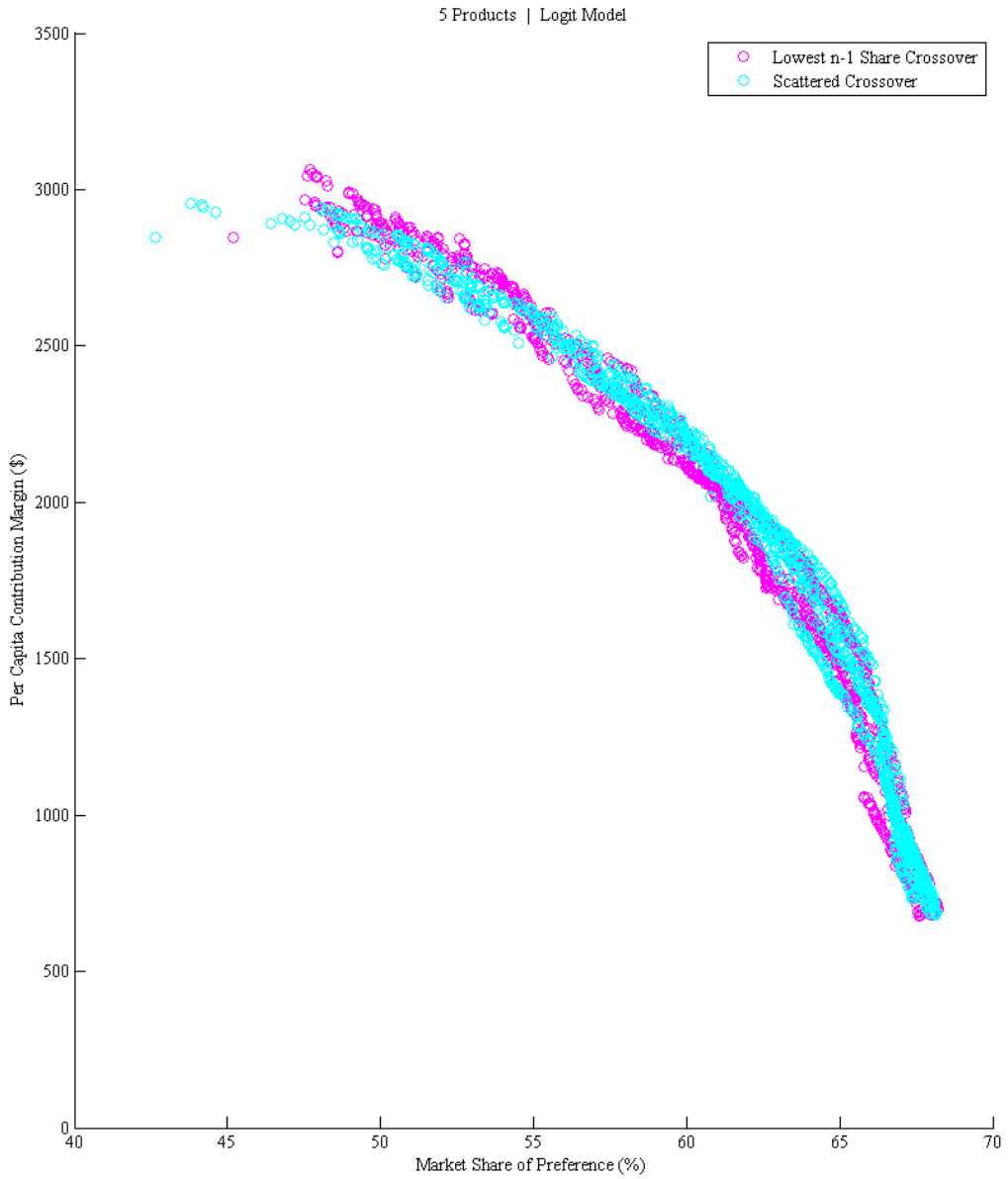

Figure B.6: Pareto Frontiers for Multi-Objective Optimization of Vehicle Product Line Configuration Using Probabilistic Choice Rule, Targeted Initialization, and 5 Products

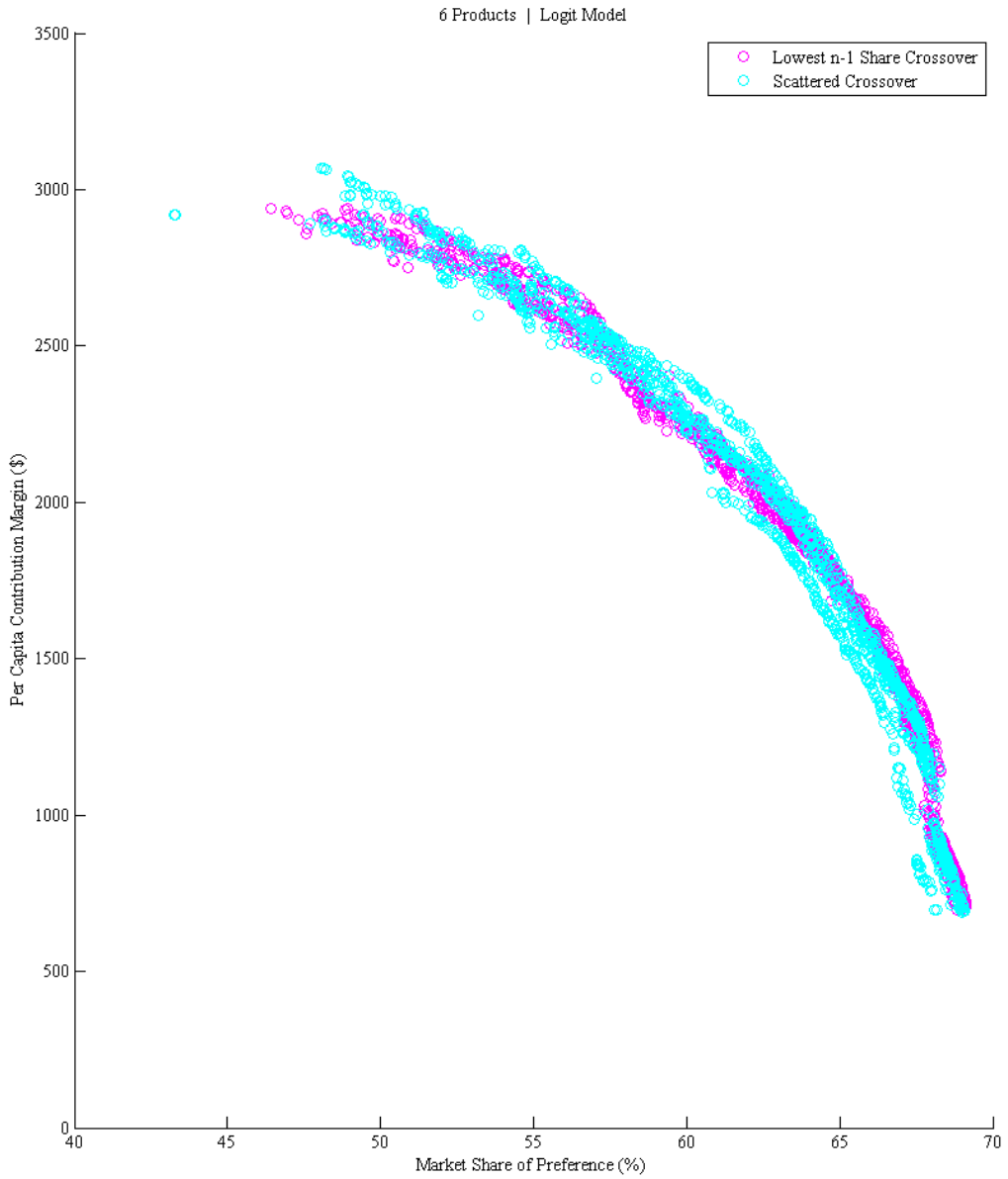

Figure B.7: Pareto Frontiers for Multi-Objective Optimization of Vehicle Product Line Configuration Using Probabilistic Choice Rule, Targeted Initialization, and 6 Products

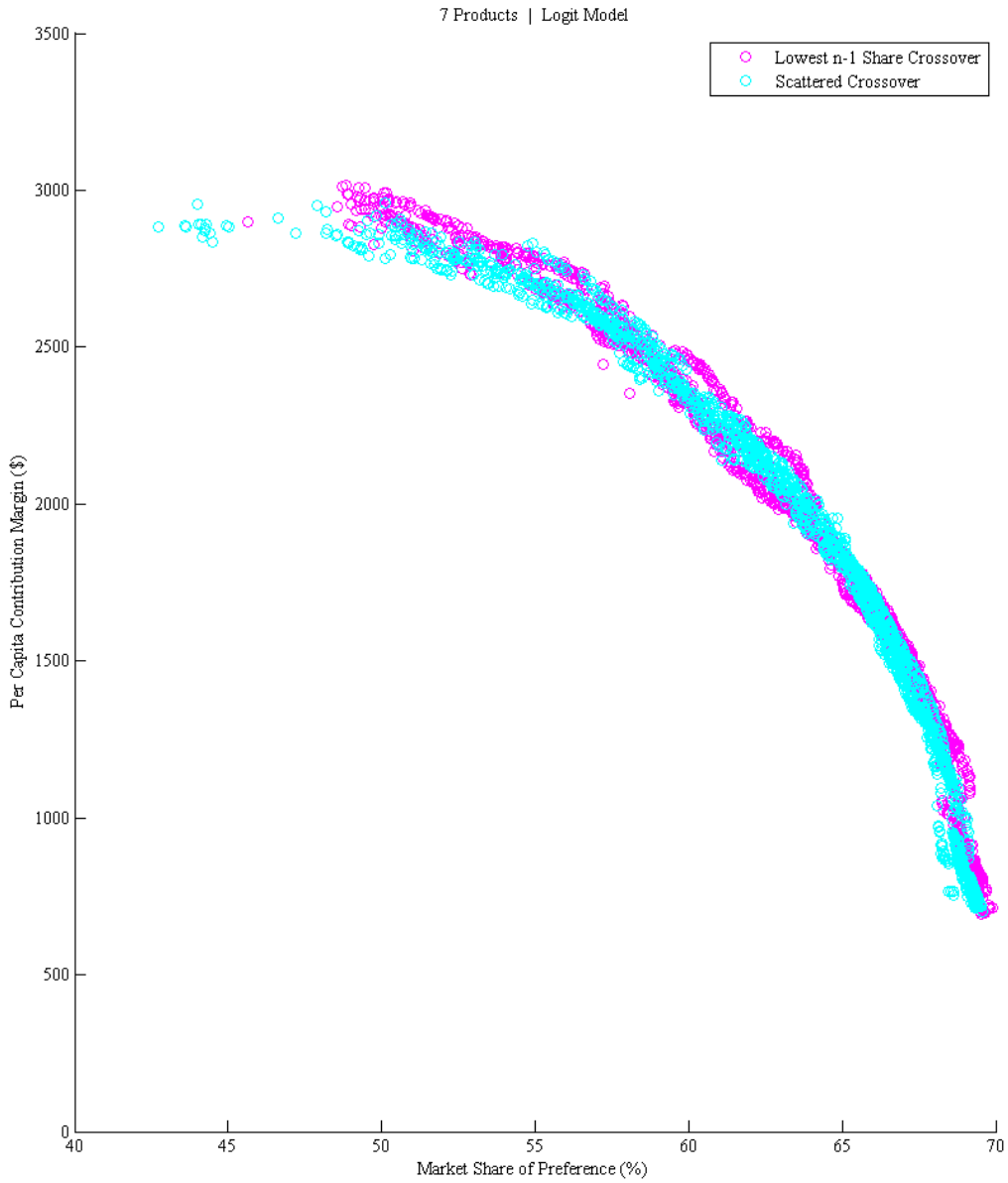

Figure B.8: Pareto Frontiers for Multi-Objective Optimization of Vehicle Product Line Configuration Using Probabilistic Choice Rule, Targeted Initialization, and 7 Products

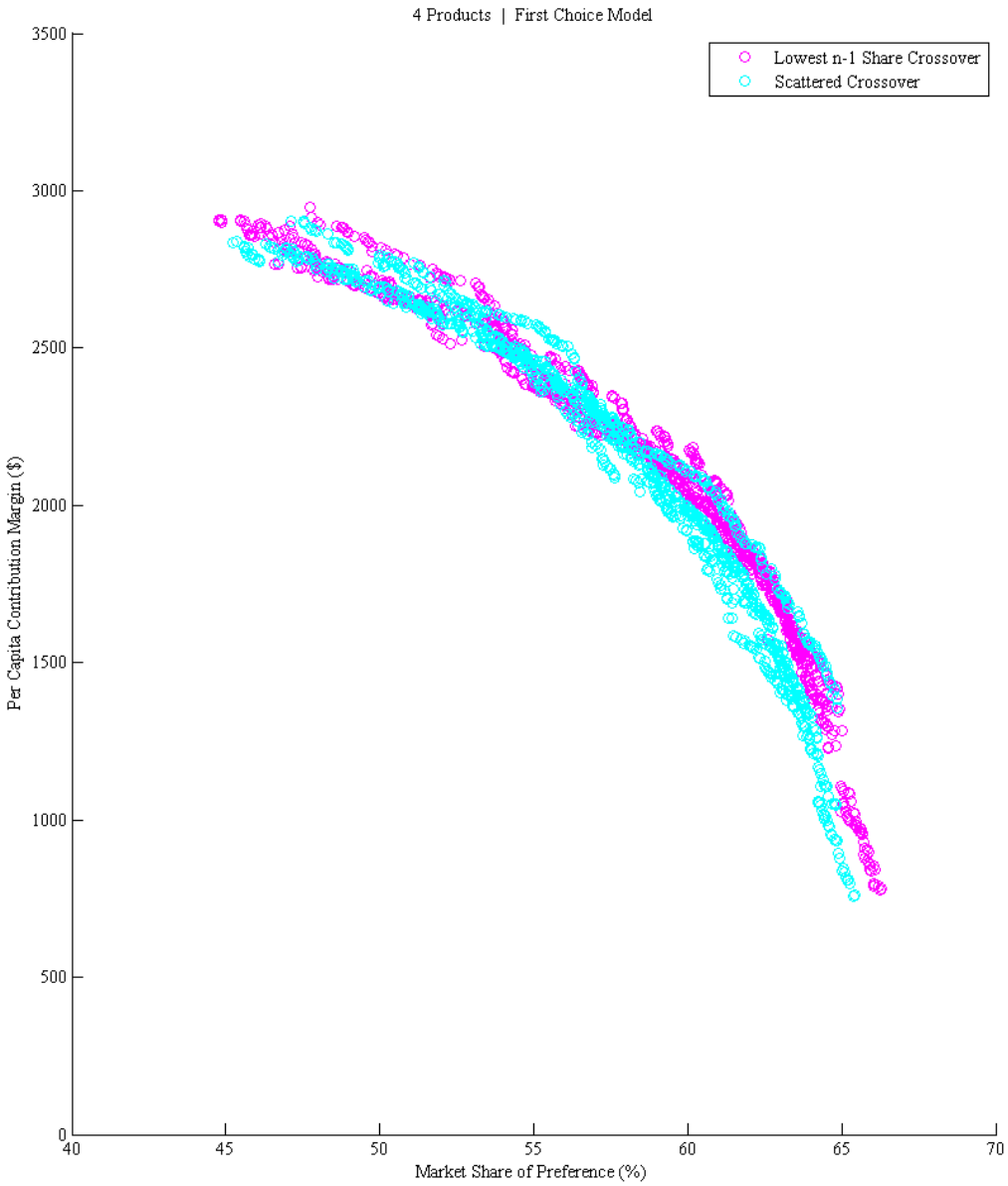

Figure B.9: Pareto Frontiers for Multi-Objective Optimization of Vehicle Product Line Configuration Using First Choice Rule, Random Initialization, and 4 Products

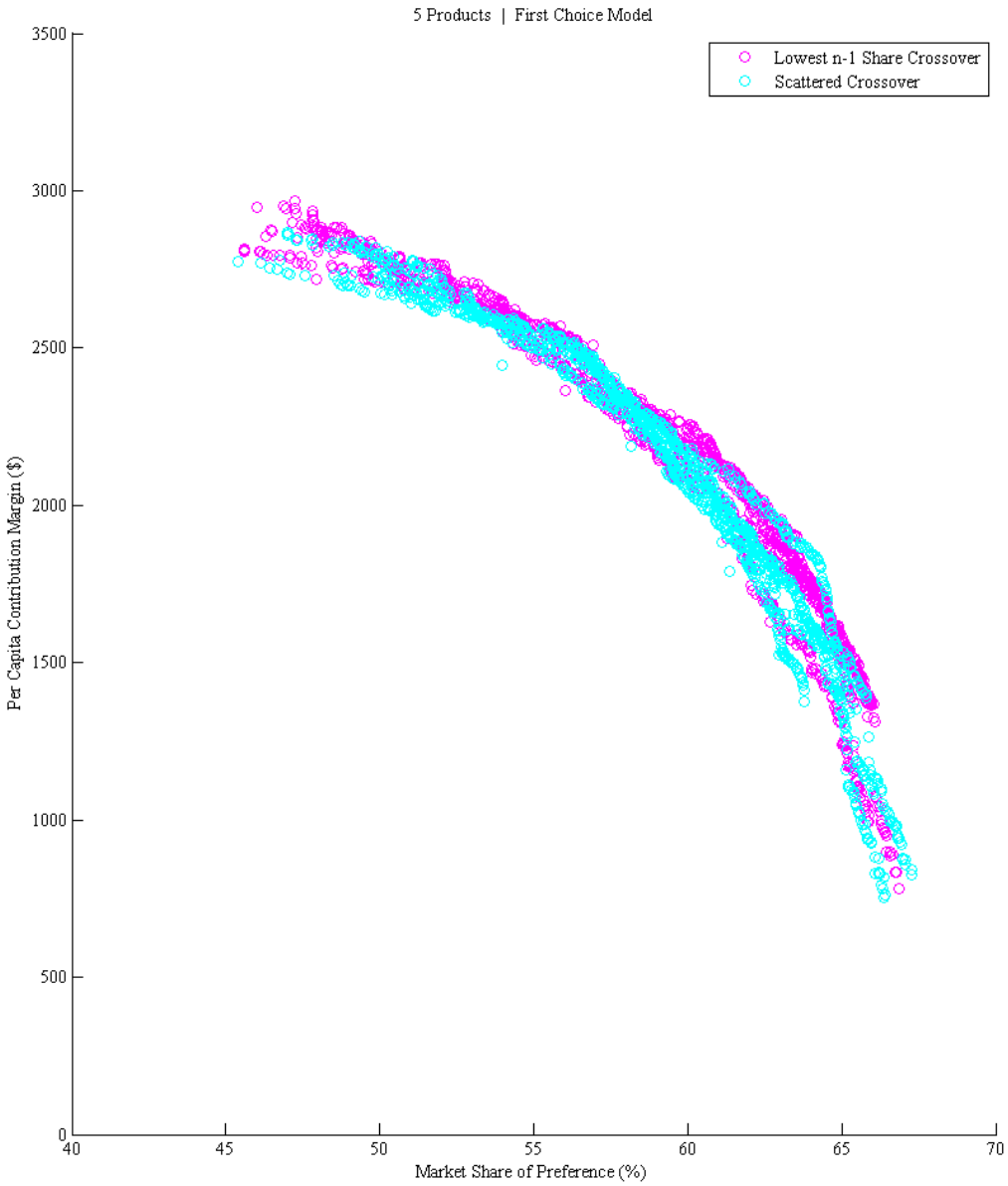

Figure B.10: Pareto Frontiers for Multi-Objective Optimization of Vehicle Product Line Configuration Using First Choice Rule, Random Initialization, and 5 Products

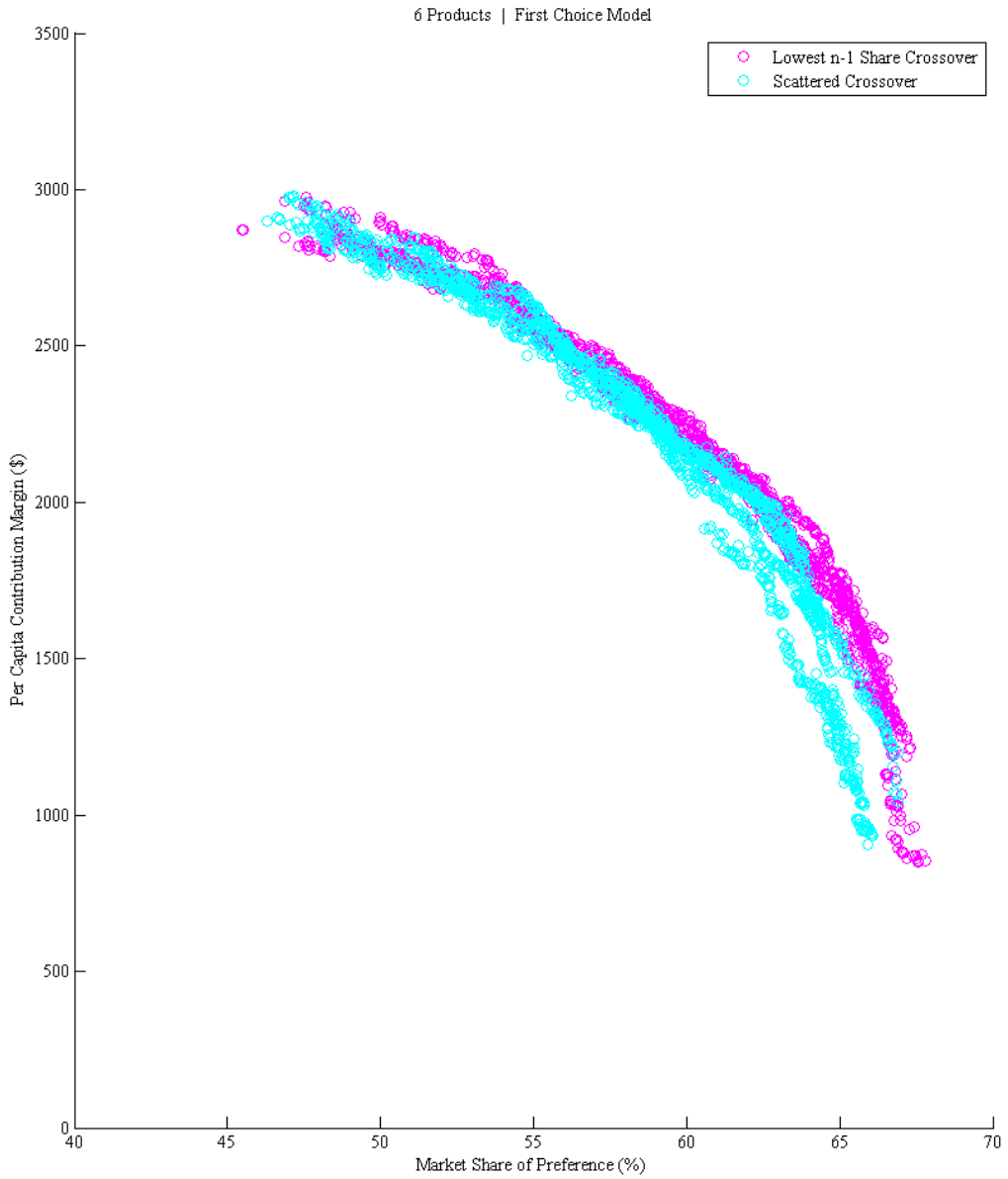

Figure B.11: Pareto Frontiers for Multi-Objective Optimization of Vehicle Product Line Configuration Using First Choice Rule, Random Initialization, and 6 Products

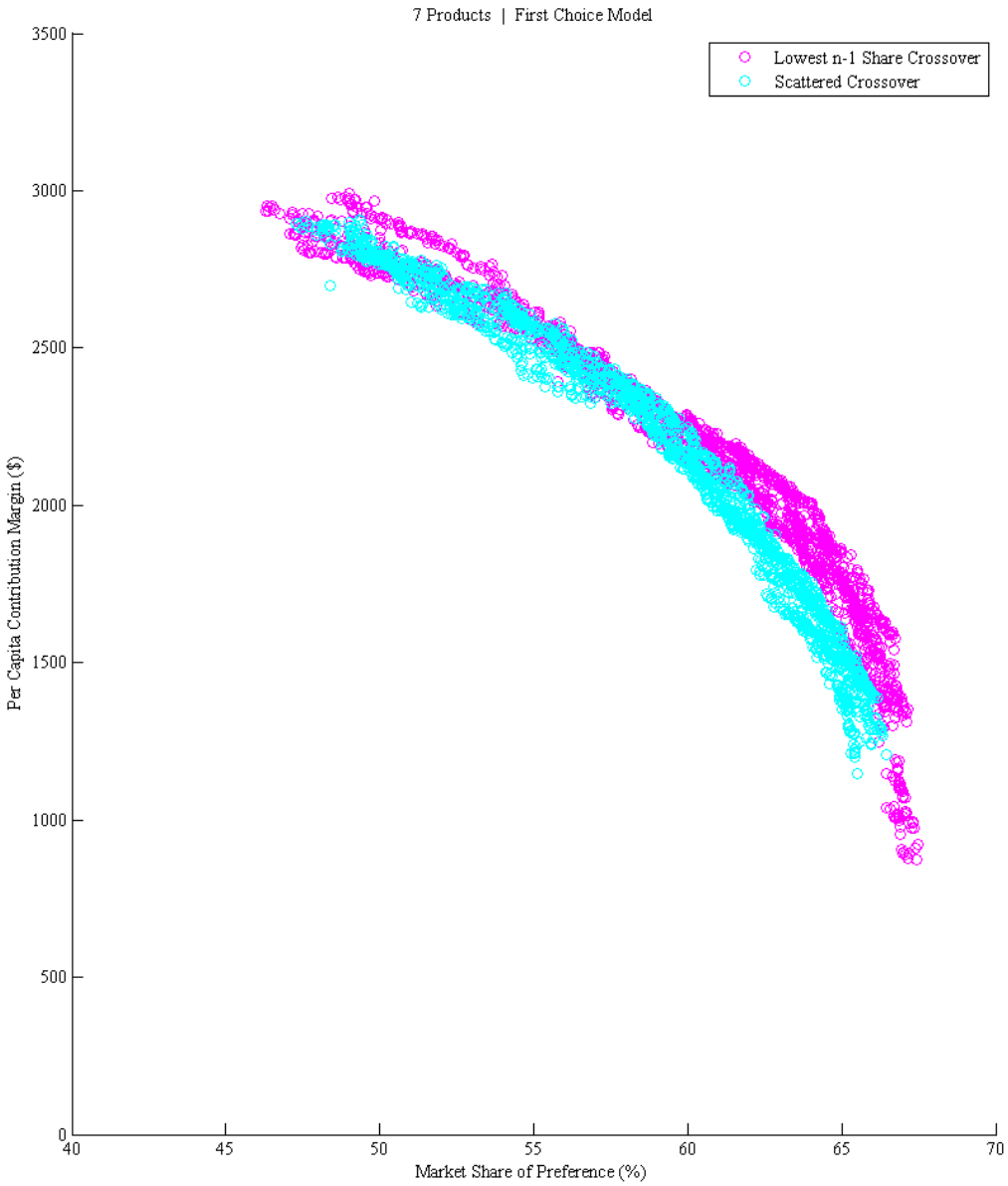

Figure B.12: Pareto Frontiers for Multi-Objective Optimization of Vehicle Product Line Configuration Using First Choice Rule, Random Initialization, and 7 Products

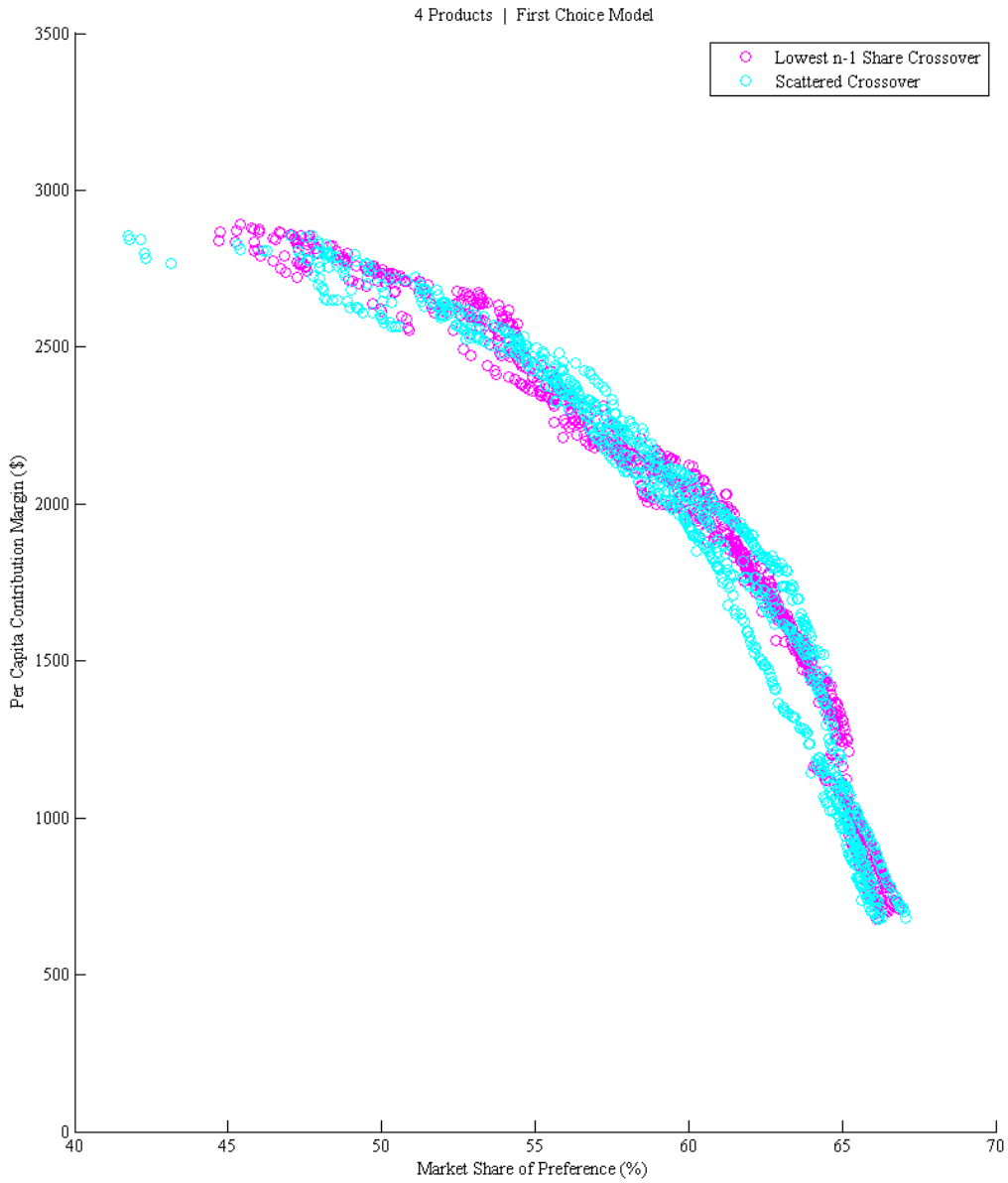

Figure B.13: Pareto Frontiers for Multi-Objective Optimization of Vehicle Product Line Configuration Using First Choice Rule, Targeted Initialization, and 4 Products

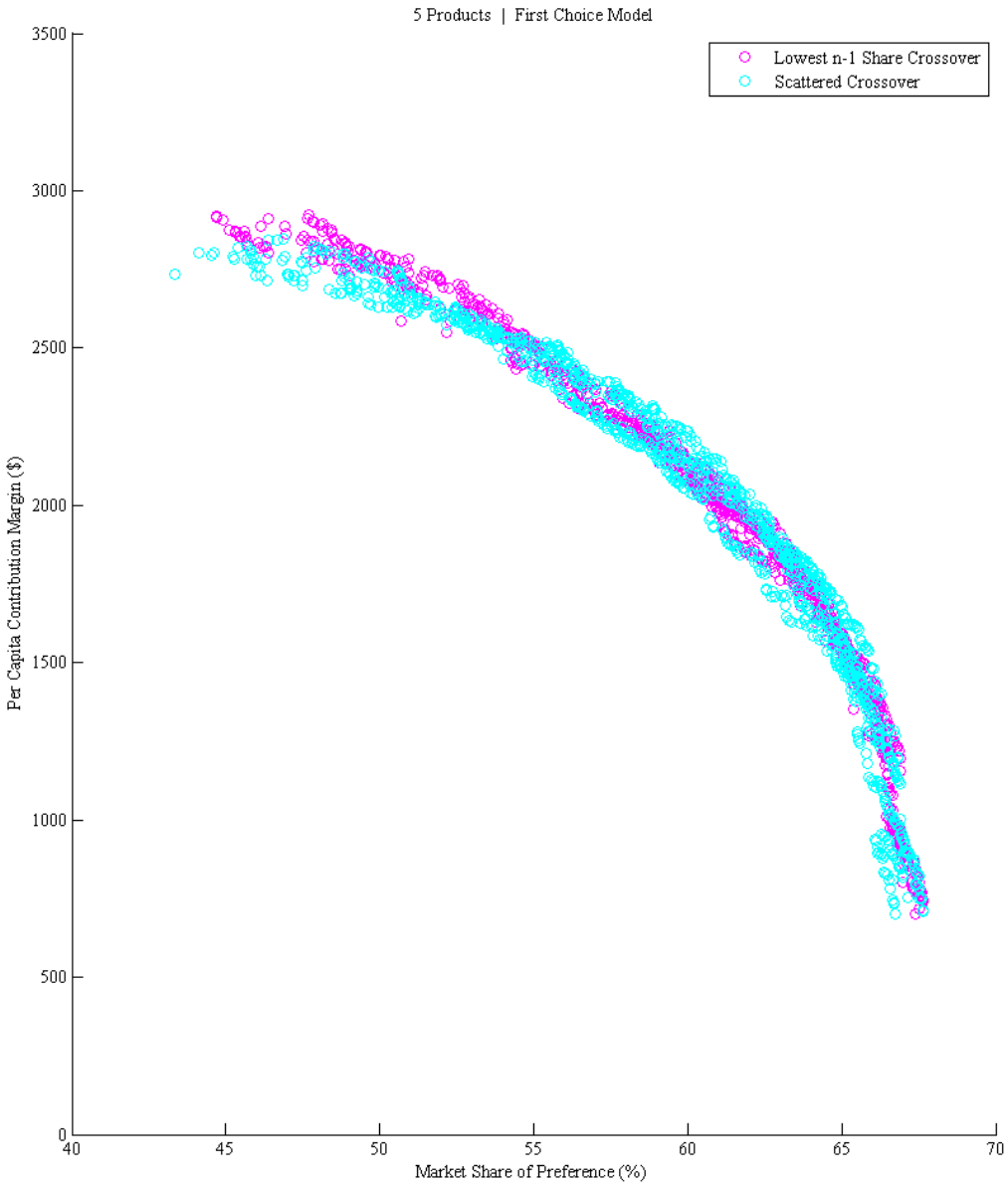

Figure B.14: Pareto Frontiers for Multi-Objective Optimization of Vehicle Product Line Configuration Using First Choice Rule, Targeted Initialization, and 5 Products

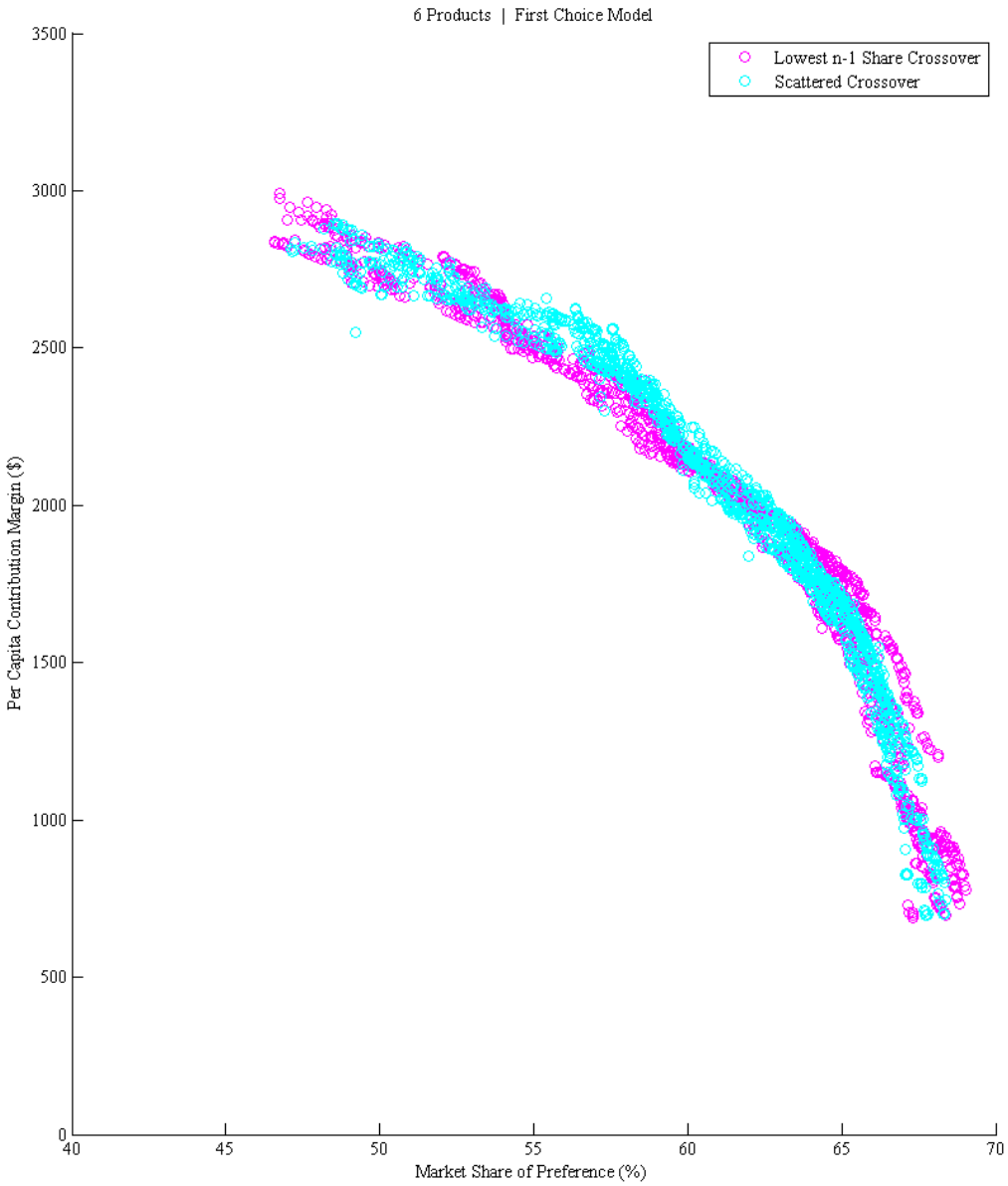

Figure B.15: Pareto Frontiers for Multi-Objective Optimization of Vehicle Product Line Configuration Using First Choice Rule, Targeted Initialization, and 6 Products

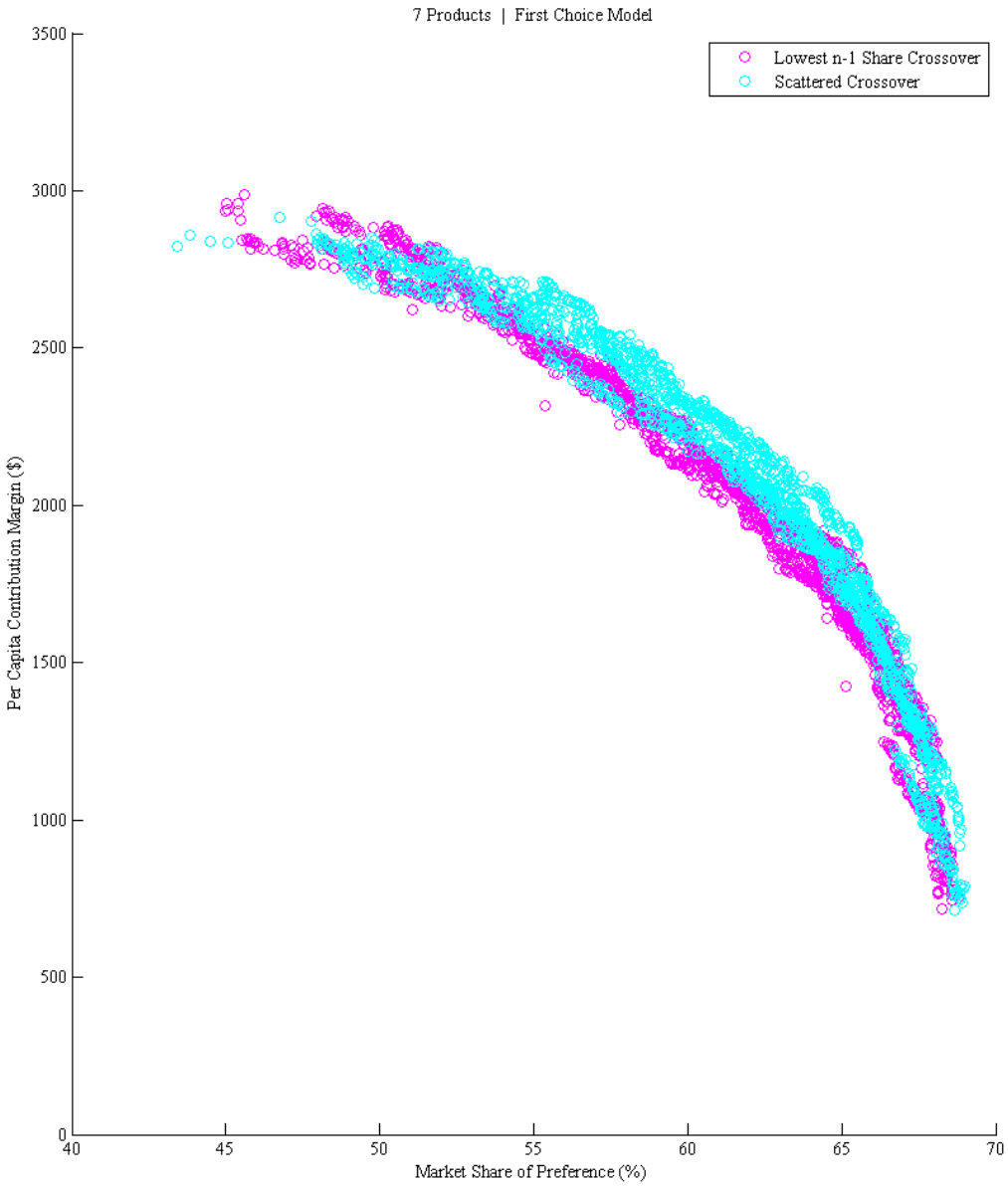

Figure B.16: Pareto Frontiers for Multi-Objective Optimization of Vehicle Product Line Configuration Using First Choice Rule, Targeted Initialization, and 7 Products