Targeted Initial Populations for Multiobjective Product Line Optimization

Callaway Turner\textsuperscript{1} and Scott Ferguson\textsuperscript{2}

\textit{North Carolina State University, Raleigh, NC, 27695, USA}

Joseph Donndelinger\textsuperscript{3}

\textit{General Motors Research and Development Center, Warren, MI, 48090, USA}

Genetic searches often use randomly generated initial populations that maximize genetic diversity to thoroughly sample the design space. While many of these initial configurations perform poorly, the tradeoff between population diversity and solution quality is typically acceptable for small design spaces. However, as the design space grows in complexity, computational savings and initial solution quality become more significant considerations.

This paper synthesizes advancements from market-based design and heuristic optimization research to strategically construct “targeted” initial populations capable of reducing computational cost and improving final solution quality. Respondent-level utilities from a discrete choice model are used with a price segmentation strategy to efficiently populate designs in a multiobjective environment where designers can explore trade-offs between competing business objectives. Results from an automobile feature packaging problem demonstrate the effectiveness of this approach, and recommendations toward the extent of price segmentation required and future research efforts are offered.

I. Introduction

Product designers face the challenge of creating products for markets with highly heterogeneous customer preferences. Further, American manufacturers must compete with manufacturers in developing nations who enjoy inherently favorable cost structures. Being both competitive and profitable requires balancing product variety and the costs of product development and production. This balance is made even more delicate as customers look for products with greater feature offerings, leading to design problems of greater complexity.

Increased problem complexity has led to two significant ramifications: 1) advanced techniques are required to capture and model customer preferences for product features, and 2) the design space associated with product line design is so expansive that traditional optimization techniques do not provide adequate solutions to the resulting combinatorial problem. To offer some perspective, automotive feature packaging problems can have hundreds of available features and hundreds (or thousands) of product variants. Additional design challenges arise from the fact that many product features cannot be represented in a continuous design space, meaning gradient-based optimization techniques cannot effectively be used. Further, since some attributes (such as price) can be represented as a continuous variable, a product designer is faced with finding optimal solutions to a mixed-integer problem.

The objective of this paper is to introduce a technique – developed by synthesizing advancements from market-based design and heuristic optimization research - that is capable of reducing computational cost and improving final solution quality while operating in a multiobjective environment where designers can explore the required market-driven tradeoffs. The rationale for this technique is described in the following sub-sections.

A. Market-based design

The application of utility and preferences as a basis for driving design decisions has been constantly evolving over the last 30 years. Work in the area of decision-based design, for example, initially focused on designer utility

\textsuperscript{1} Graduate Research Assistant, Department of Mechanical and Aerospace Engineering, Student member AIAA  
\textsuperscript{2} Assistant Professor, Department of Mechanical and Aerospace Engineering; scott_ferguson@ncsu.edu, Member AIAA  
\textsuperscript{3} Staff Research Engineer, General Motors Research and Development Center; joe.donndelinger@gm.com, Member AIAA
for engineering options, but quickly evolved to include consumer utilities. Solution quality in this framework focused on maximizing the net present value of profit, as demonstrated by Mistree and Marston.

Simultaneously, engineering design methods leveraging market-demand models began to mature, beginning with the utilization of conjoint analysis and the S-Model. Li and Azarm, for example, built upon Hazeltineg’s framework by using conjoint analysis and regression to fit utility functions at the respondent level. This work also extended the scope of the business domain by considering net present values of both market share and profit.

Choice-based conjoint studies and discrete choice analysis ushered the next evolution with the added level of realism associated with selecting from a set of alternatives. These works saw the first application of the logit model, experimental methods for profiling the market and mapping to the technical space, exploration of model assumptions and their implication on results, integration with existing design-decision tools, exploration of market heterogeneity, application toward mass customization, and the effect of retail channels. However, these works focused on single product design.

The focus on product line optimization can be traced back to Green and Krieger, who formulated the problem around selecting from candidate products to maximize buyer welfare. As with single product design, conjoint analysis was the first extension to product line design. A nested-logit model was also used by Kumar et al. to design a product family for four hypothetical market segments.

As research advanced to discrete choice analysis, demand-model formulation progressed from a multinomial logit to more advanced forms capable of representing customer heterogeneity. These works have primarily explored the latent-class multinomial logit and hierarchical Bayes mixed logit formulations.

The evolution of research in this area has highlighted the rich heterogeneous preference information captured by hierarchical Bayes mixed logit models using data gathered from choice-based conjoint surveys. While many choice model forms can be used in engineering design, research has demonstrated that for product line problems with heterogeneous customers, hierarchical Bayes has clear advantages. The richness from this model form is necessary to adequately represent the tradeoff decisions that must be made by a designer in the technical and marketing domains. However, this richness comes at a price, as the resulting optimization problem can now require tens or hundreds of design variables if the entire product line is to be simultaneously optimized.

### B. Product line optimization approaches

Early efforts in product line optimization relied on greedy heuristic approaches and other heuristic rule-based approaches. Analytical methods, like linear programming and ATC, have also been used with multinomial logit models with reasonable success. However, such approaches are commonly not suitable for problems with discrete attribute levels.

To accommodate discrete and mixed-integer problem formulations, an increasing amount of research has turned to heuristic optimization techniques like branch-and-bound and genetic algorithms. While initial adoption of heuristic techniques was used to solve single product problems, these techniques have since been extended to product line optimization and have been shown to be more effective than greedy algorithms and analytical approaches.

Genetic searches have been shown to be easy to set up, perform well for these problem types, and the influence of genetic search parameters (crossover and mutation) on solution quality has undergone preliminary investigation. However, generalized tailoring procedures involving the initial population are mainly an unexplored area of research. Most genetic search research advocates the use of random draws, leading to algorithms that ensure diversity amongst the initial designs. For mathematical test problems and most engineering optimization scenarios, where the structure of the final solution is unknown, this diversity is desired. However, as the design space becomes more complex, reducing the number of required function calls becomes a priority. Balakrishnan et al., for example, saw solution quality improvement when they seeded their initial population with results from a Dynamic Programming heuristic. Yet, the arbitrary nature of the sequencing algorithm allowed for good solutions to be discarded early in the process.

Beyond this work, however, there has been little effort to improve optimization algorithms for the market-based design of discrete combinatorial module-based products. Because of this, the rich source of information present in the heterogeneous part-worth utilities obtained from the demand-model remains largely untapped. We argue that this information can be used to gain insight into the designs preferred by the individuals who took part in the choice-based conjoint survey. For single objective problems, prior work by the authors has demonstrated the effectiveness of intelligently seeding a genetic search using these preferred designs. Details of this approach are described in Section II. However, this work also demonstrated that when multiple objectives are considered, further extensions to this approach are necessary.
C. Multiobjective product line design

While the DBD framework proposed by Hazzelrigg champions the maximization of net present value of profit, this formulation is only possible if the decision-makers can fully articulate all required information. Balling argues that this often is not possible, and that decision-makers may articulate - and update - their preference structure only after “shopping” the feasible solution space. This “design by shopping” paradigm has supported the creation of tradespace visualization tools and multiobjective problem formulations.

In market-based product design, Besharati et al. used a multiobjective genetic algorithm to explore the tradeoff between share and share variation due to uncertainty. Multiobjective formulations for product line design occur most often in the context of product platforming. Tradeoffs of interest can include platform performance versus product commonality and design objectives versus market share.

If all information from the decision-makers cannot be fully articulated, companies may not wish to establish their market positioning strategy based only on share of preference. Often preference share may be easily increased by reducing prices across the entire product line; however, this action typically has a negative impact on overall profitability. Thus there is an inherent tradeoff between preference share and profit that must be explored. Yet, formulating a market-based product line design problem in this context has not been previously investigated.

Efficiently populating the Pareto frontier for a product line design problem will be significantly aided by approaches that reduce the computational cost of the multiobjective genetic search. Prior work by the authors demonstrated that synthesizing research advancements from the areas of market-based design and product line optimization led to significant reductions in computational cost and improved solution quality for single objective formulations. This paper further extends the use of an intelligent initial population (referred to as a “targeted population”) to multiobjective problem formulations in support of the “design by shopping” paradigm.

The remaining sections of this paper are as follows. The technical foundation for the targeted population is introduced in Section II. Extension of this technique to support multiobjective problem formulations is presented in Section III. Results from a case study problem are then presented in Section IV, with conclusions and future work discussed in Section V.

II. Technical foundation for targeted populations

Creating an initial population using random draws can lead to technically feasible starting designs that are also absurd, guaranteeing a very low probability of marketplace success. This strategy has come to be accepted because there is rarely insight into how to tailor the starting population by including product characteristics known to be desirable. However, when demand-model information is integrated with product design, a customer’s part-worth utilities for each attribute level are known. These part-worth utilities give product designers a rich source of information that can be used to identify preferred designs.

A. Finding the preferred design

The objective of this approach is to create initial product lines by drawing from a pool of products that are optimal for at least one respondent. It is not guaranteed that each optimal product will perform well at the market level or capture new pockets of market share in a product line scenario. However, it is reasonable to expect that if at least one respondent finds a product highly desirable, it could be an effective starting point for the genetic search.

When heterogeneous models are used in product design, a respondent’s utility for an alternative is represented by decomposing it into an observable component \( V_{nj} \) and an unobservable component \( \varepsilon_{nj} \) as in Eq. 1.

\[
U_{nj} = V_{nj} + \varepsilon_{nj} \quad j = 1 \ldots J
\]

The observable component of utility \( V_{nj} \) - or representative utility - is itself expressed as a function of alternative attributes \( x_{nj} \), coefficients of the alternative attributes \( \beta \), and consumer attributes \( s_n \). Coefficients of the alternative attributes are estimated statistically using respondent choice data from a choice-based conjoint survey. To obtain this choice data, respondents completed a choice-based conjoint survey generated by Sawtooth Software’s CBC Web tool. Part-worths for each individual are then estimated using Sawtooth Software’s CBC/HB module. While we use the results from CBC/HB in this work, this approach is generalizable to any method capable of determining utility functions for each individual.

Respondent-level optimal products are found by performing a series of sub-optimizations to identify the attribute level settings that maximize observable utility. It is also necessary to assign price structures for each attribute;
otherwise no tradeoffs would be necessary and the ideal product would simply consist of the most preferred level of every attribute at the lowest surveyed price point. Formulation of the optimization sub-problem is shown in Eq. 2.

\[ \text{Max : } V_{nj} = f(x_{nj}, \beta_n, s_n) \]

with respect to : \( x_{nj} \), price of including \( x_{nj} \)

Sub-optimizations are typically completed very quickly and the cost of the optimization is relatively small because the size of the design space is greatly reduced. Further discussion on computational cost for this approach can be found in Turner et al.\textsuperscript{47}.

B. Generating initial product lines

The pool of ideal products provides source material for the initial product lines. While the number of ideal products is theoretically limited by the number of survey respondents (\( n \)), the size of the initial population must be defined by the designer. The literature suggests an initial population size approximately 7-10 times the number of design variables being considered\textsuperscript{58}.

As shown in Fig. 1, the ideal products created in the previous sub-section are randomly selected, without replacement, and combined to create product lines. The remainder of the product line optimization is now conducted using standard genetic search operators.

![Figure 1. Illustration of creating targeted initial product lines](image)

C. Single objective optimization results

The original problem motivating this research is based on a choice-based conjoint study commissioned by General Motors, in which 19 choice task questions were answered by 2275 respondents. Nineteen product attributes were considered in this study, yielding over 1,074,954,240 possible product configurations that must be considered.

The number of unique initial product line configurations was set at 100 times the product line size. This corresponds to approximately 10 times the number of design variables\textsuperscript{58}. For comparison, the results from two approaches are shown in Fig. 2 and Fig. 3. These approaches correspond to the targeted initial population and a traditional initial population created using random draws. The best intermediate solutions at an equivalent number of objective function evaluations from 5 runs of each approach are shown in Fig. 2. This snapshot accounts for the initial overhead required to generate the targeted initial population.

When both approaches are run to completion, the random population returns solutions similar to those found using the targeted population. This is shown in Fig. 3. However, a significant advantage of the targeted population is reduction of computational cost, as shown in Table 1.

<table>
<thead>
<tr>
<th>Number of products</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
</tr>
</thead>
<tbody>
<tr>
<td>Random population</td>
<td>12400</td>
<td>23400</td>
<td>40800</td>
<td>149000</td>
<td>100200</td>
<td>141400</td>
<td>162400</td>
</tr>
<tr>
<td>Targeted population</td>
<td>6800</td>
<td>13500</td>
<td>28800</td>
<td>52236</td>
<td>59836</td>
<td>129636</td>
<td>83036</td>
</tr>
</tbody>
</table>
This section introduced the technical foundation for creating a targeted population and demonstrated the effectiveness of the approach for a single objective problem. As the motivation for this work is to facilitate “design shopping” of product line solutions in a multiobjective problem formulation, the extension of this approach is discussed in the next section.

III. Extending targeted populations to multiobjective problem formulations

When a pricing structure for feature attributes is defined a priori in a product design problem, the market-based objectives of profit and share of preference are linked in a cooperative manner. This is because the marginal return on a product has already been established; turning the design problem into one where maximizing the share of preference will correspondingly maximize the company’s profit. The targeted initial population approach described in the previous section is well-suited for this type of problem formulation. By creating an initial population with designs that are highly desirable to individual customers, the assumption is that this provides a more effective starting point for a genetic search tasked with maximizing market share of preference.

However, companies may not wish to determine their market positioning strategy based solely on share of preference. Rather, companies may wish to more deeply explore the inherent tradeoffs that exist between preference share and profit when defining their corporate strategy. Providing this capability requires relaxing the assumption that an a priori pricing structure for feature attributes has been defined. However, relaxing this assumption not only facilitates a richer multiobjective problem, but also increases the complexity of the problem to be solved.

When a price structure is defined prior to searching the design space, the price associated with an attribute level can be treated as a constant, as shown in Table 2. When a product is fully specified, the total price is simply the summation of constants associated with active attribute levels. However, if the price of an attribute level is not pre-defined, it becomes a design variable that must be specified by the search algorithm. For this problem formulation, bounds on feature level price can be established if market price thresholds are known, as demonstrated in Table 2.

Table 2. Representative price structure for a hypothetical product attribute

<table>
<thead>
<tr>
<th>Attribute level</th>
<th>Pre-defined price structure</th>
<th>Price as design variable</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Low price bound</td>
</tr>
<tr>
<td>1</td>
<td>$10</td>
<td>$0</td>
</tr>
<tr>
<td>2</td>
<td>$90</td>
<td>$60</td>
</tr>
<tr>
<td>3</td>
<td>$185</td>
<td>$120</td>
</tr>
</tbody>
</table>

The computational impact of this new problem formulation is not insignificant, as the number of design variables increases to accommodate this freedom. This increase in problem complexity is illustrated in Fig. 4 for the design of a single product. In this figure, each product attribute is represented by a series of boxes, which corresponds to the number of levels that embody that attribute. A binary encoding structure is used within each attribute to designate the level of the attribute that is active in the current design.

For a pre-defined price structure, the price of the product is simply the summation of the individual price components. However, when price structure is not defined a priori, the variable increase is the summation of all attribute levels considered in the choice-based conjoint study.
When this approach is extended to product line design, the computational penalty is proportionally less severe. This is because it is assumed that attribute level prices are constant across the entire product line. While this assumption could be relaxed in future work—particularly to facilitate product family design across multiple market segments—we treat the product line as variants that exist within the same market segment. Therefore, while the number of design variables increases with each product added to the line, the computational penalty associated with including attribute level price as a design variable is only assessed once.

A. Formulating the multiobjective problem

In this work, two objective functions are simultaneously considered. The first objective involves maximizing the share of preference captured by the product line. This is identical to the objective function used to introduce the targeted population in Section II. The second objective involves maximizing the surrogate of a metric known as aggregate contribution margin (ACM). Contribution margin is the per unit difference between a product’s selling price and its cost; it does not account for investments, other fixed costs, or the time value of money. This metric was strategically selected, as it only contains terms (product price and product cost) that are directly controlled by a designer. For our proxy of ACM, this measure is multiplied by share and aggregated across all products in the line, as shown in Eq. 3. The number of products being simultaneously designed is given by $n$.

$$ACM \propto \sum_{i=1}^{n}(price_i - cost_i) \times share_i$$  \hspace{1cm} (3)

Adding a surrogate for ACM as an objective function introduces tension between the customer-based objective of maximizing probability of choice and the company-based objective of maximizing profit. In this work, pricing structures have been established for each attribute level based on costs and market price thresholds determined by subject matter experts at General Motors. The representative pricing structure shown in Table 3 for a hypothetical product attribute is designed to illustrate how the attribute-level contribution is directly influenced by the cost and price value set for each attribute level by the search algorithm. The full multiobjective problem formulation is shown in Eq. 4.

<table>
<thead>
<tr>
<th>Attribute level</th>
<th>Cost</th>
<th>Low price bound</th>
<th>High price bound</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>$0</td>
<td>$0</td>
<td>$20</td>
</tr>
<tr>
<td>2</td>
<td>$50</td>
<td>$60</td>
<td>$120</td>
</tr>
<tr>
<td>3</td>
<td>$120</td>
<td>$120</td>
<td>$250</td>
</tr>
</tbody>
</table>

Table 3. Representative price structure for a hypothetical product attribute
Maximize: \( \text{Preference share ACM} \) \hspace{1cm} (4)

with respect to: \( \text{Feature content Feature price} \)

subject to: \( \text{Feature price bounds} \)

**B. Promoting multiobjective diversity in the targeted population**

Recalling Eq. 2, generating candidates for the targeted initial population involved finding the product with the greatest observed utility \( (V_{nj}) \) for each respondent. When a pricing structure is pre-defined, this optimization problem often becomes one of balancing increased features, increased functionality, and/or increased performance against the corresponding increase in product price. The reason for this tradeoff is shown in Fig. 5, which shows a single respondent’s part-worths determined for each price level seen in a choice-based conjoint survey. As price increases (going from attribute level 1 to attribute level 12) the part-worth value decreases. A working assumption when determining the part-worths for each attribute level is that consumer behavior is compensatory; that is, a decrease in observed utility because of a change in attribute level can be offset by an equal increase in observed utility because of a change in another attribute’s level. Therefore, a change in level for one attribute (ex: going from ‘good’ to ‘better’) likely causes a change in price (ex: going from $ to $$) that may result in a net utility change of zero. The objective of the optimization shown in Eq. 2 can be simply understood as finding the product configuration where the net compensation is as large as possible.

![Figure 5. Part-worth value for price levels for a single respondent](image)

When attribute price is treated as a design variable, the compensatory nature of the observed utility calculation motivates the need to modify how candidate designs for the targeted initial population are generated. If the problem statement in Eq. 2 is used, and if an increase in price leads to a decrease in the price part-worth, the optimum price variable for each attribute level will always be set to the low bound. From a multiobjective optimization standpoint, the ramifications of this outcome are significant, as shown in Fig. 6. In this figure, each candidate design is simulated against an outside good in a market simulator for all survey respondents. Note that the ACM axis has been scaled to protect sensitive information from the industry partner.

To provide perspective, candidate designs generated with attribute price variables fixed at the upper bound are also shown. From these two clusters of designs, interesting insights can be drawn. First, candidate designs created with all attribute price variables set at the lower price bound have a noticeably richer representation along the share of preference axis. Second, these designs do not populate as wide of a range along the ACM axis as their upper price bound counterparts. Of possibly even greater significance is the number of designs located below the x-axis. These designs represent products that could feasibly be created and sold; however, the cost of making the product is greater than the sale price.

Conversely, all candidate designs generated with attribute price variables fixed at the upper bound have a positive ACM value. However, while these designs have a richer representation along the ACM axis, the increase in attribute prices causes a more significant diversion to the outside good. Although fewer of these products are sold, the increased margin for these products can lead to improved profit. These results not only demonstrate the competitive relationship between the consumer and business objectives (share vs. ACM), but also provide insight into how the procedure for generating targeted initial populations can be modified for multiobjective problems.
For single objective problems, an optimization algorithm can be considered to be more effective if a solution is found that further minimizes (or maximizes) the objective and/or if the optimum can be found in fewer function calls. When solving multiobjective optimization problems, however, quality of solution can be defined by multiple metrics\textsuperscript{59}. In this work, the goal of using a targeted initial population is to support the “shopping” of the performance space by finding a Pareto frontier that 1) populates regions not previously captured, 2) dominates the original solution, 3) reduces computational cost, and 4) identifies a greater number of Pareto points.

As shown in Fig. 6, greater multiobjective diversity in the targeted population can be gained by considering different attribute price points. Therefore, to capture a more complete representation of the solution space, candidate designs for the targeted initial population will be generated using multiple attribute price points. This can be accomplished by using the lower and upper price bounds for each attribute level and applying the same markup value – a number between 0 and 100 that is interpolated along the price scale. When multiple price points are defined, they are evenly spaced between the price bounds, as shown in Table 4.

\begin{table}[h]
\centering
\begin{tabular}{|c|c|c|c|c|}
\hline
Attribute & Price Range & Price Point 1 & Price Point 2 & Price Point 3 \\
\hline
Attribute 1 & Level 1 & $0 - $20 & $0 & $10 & $20 \\
& Level 2 & $60 - $120 & $60 & $90 & $120 \\
& Level 3 & $120 - $250 & $120 & $185 & $250 \\
Attribute 2 & Level 1 & $10 - $40 & $10 & $25 & $40 \\
& Level 2 & $100 - $150 & $100 & $125 & $150 \\
\hline
\end{tabular}
\caption{Establishing price points for hypothetical product attributes}
\end{table}

Once the number of price points has been established, candidate designs for the targeted population are generated for each respondent at each price point. The targeted initial population is then populated using a proportional number of candidate designs from each price point. For example, if three price points are used, a third of the targeted initial population will come from each price point.

Candidate designs are then randomly selected from the same price point and combined into product lines. Aggregation is done in this manner so that the same pricing structure is applied to all products within a given product line. Once the initial product lines are evaluated, they undergo the operations of crossover and mutation with other product lines. This allows for changes to design variables associated with both content and price structure, as shown in Fig 7.
Figure 7. Targeted initial population creation procedure for multiobjective problems

Having described the manner by which the targeted initial population is created for multiobjective problems, the next section of this paper demonstrates the application using a case study problem.

IV. Case study problem

The case study for this paper is a feature packaging problem for an automobile product line. Section II highlighted the advantages of the targeted initial population for this problem when a single objective was considered. This problem has 2275 survey respondents, nineteen product attributes, and the number of levels associated with each attribute is shown in Table 5. Attribute levels represent a feature’s presence / absence or the various means by which that feature can be implemented. Enumeration of all possible feature combinations yields over 1,074,954,240 product configurations that must be considered. A line of five products is to be created.

Table 5. Levels per attribute for the case study problem

| x1 | x2 | x3 | x4 | x5 | x6 | x7 | x8 | x9 | x10 | x11 | x12 | x13 | x14 | x15 | x16 | x17 | x18 | x19 |
|----|----|----|----|----|----|----|----|----|-----|-----|-----|-----|-----|-----|-----|-----|-----|
| 3  | 2  | 5  | 6  | 2  | 3  | 3  | 2  | 4  | 2   | 3   | 2   | 4   | 3   | 4   | 4   | 3   | 2   |

A pricing structure similar to Table 3 is used for this problem; however, because of its proprietary nature it cannot be disclosed in this paper. To provide a baseline solution, a MOGA is run using a randomly generated initial population. This algorithm was coded by the authors in Matlab, and the settings used for the optimization are shown in Table 6. Effective convergence criteria for multiobjective optimization are still a current topic of research, so results comparisons will be made at equivalent numbers of objective function evaluation. For this case study, the reporting of function evaluations signifies the effort to evaluate a product line solution across all objective functions considered. That is, one evaluation is the total effort to calculate the product line’s preference share and ACM. Figure 8 illustrates the progression of the Pareto frontier when the algorithm is seeded with a randomly generated initial population.

Table 6. MOGA input parameters

<table>
<thead>
<tr>
<th>Criteria</th>
<th>Setting</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pool size</td>
<td>(10^n(#DVs))</td>
</tr>
<tr>
<td>Selection</td>
<td>Random</td>
</tr>
<tr>
<td>Crossover type</td>
<td>Uniform</td>
</tr>
<tr>
<td>Crossover rate</td>
<td>0.6</td>
</tr>
<tr>
<td>Mutation type</td>
<td>Gaussian</td>
</tr>
<tr>
<td>Stop after</td>
<td>User defined number of evaluations (MOGA)/</td>
</tr>
</tbody>
</table>

American Institute of Aeronautics and Astronautics
Having defined the problem and a baseline solution for comparative purposes, the first step in generating a targeted initial population involves defining the number of price points to be considered. This decision is explored in the next sub-section.

A. Defining price point granularity

The results in Fig. 6 showed that multiple price points are necessary to provide a richer sampling of the design space. However, the number of price points needed to ensure a diverse sampling is likely to be problem dependent. The challenge is balancing the increase in solution diversity with the computational cost associated with creating candidate designs for the targeted population. If too few price points are sampled, important regions of the design space may not be explored. If too many are sampled, the computational cost may outweigh the value of the information gained.

An approach to account for the computational overhead associated with the targeted initial population is discussed in prior work 47. To explore the effect of number of price points studied, three different cases will be investigated:

- 1 price point;
- 7 price points;
- 30 price points

When only one price point is considered (defining the mark-up parameter at 50%) each respondent has their own targeted design solution. However, as more price points are included it is not guaranteed that, at the respondent level, each price point will map to a unique product configuration. As shown in Fig. 9, when seven price points are considered, less than 500 of the 2275 respondents have seven unique targeted product configurations. Interestingly, 42 of the respondents have only one unique product configuration. For these respondents, product configuration is completely independent from the value of the attribute price variables (within the upper and lower bounds). While this result deserves further exploration, it is likely that these respondents are either extremely price sensitive (the preferred configuration is always the cheapest possible combination) or extremely price insensitive (the attribute part-worths strongly dominate the price part-worths).

If candidate designs are created for 30 price points, as shown in Fig. 10, a different trend is apparent. While the histogram in Fig. 9 is left-skewed, the histogram shown in Fig. 10 is right-skewed. No respondents have 30 unique targeted product configurations, and only 11 of the 2275 have 25 or more unique product configurations. Further, the peak of Fig. 10 occurs at 8 and 9 products. These results imply that selecting 30 price points will yield diminishing returns in terms of solution diversity, and that the computational expense of generating such a large number of candidate designs may be more effectively spent within the operators of the MOGA.
B. Solution using one price point

By treating price as a design variable, the multiobjective optimization problem contains a total of 155 design variables – 95 of which represent the 19 attributes on each of the 5 products, plus an additional 60 attribute price variables. Using the mapping relationship defined in previous work\(^7\) to determine the computational expense of creating the targeted initial population, it was calculated that 439 evaluations were required. Figure 11 shows the progression of two Pareto frontiers at 50,000, 100,000, 250,000 and 500,000 evaluations. One frontier represents the solution from the MOGA run with a randomly generated initial population. The second frontier represents the solution from the MOGA run with a targeted initial population constructed using one price point.

At 50,000 evaluations, the solution using the randomly generated initial population dominates the targeted population solution and has greater solution diversity across both axes. By 100,000 evaluations, the targeted population solution dominates the random population solution at larger preference shares (above 55\%) and has increased solution diversity along the ACM axis. The targeted population solution dominates the random population solution almost entirely by 250,000 evaluations, and at 500,000 evaluations has a larger hyperarea while also dominating the random population solution.
Both the random initial population and the targeted initial population start with the same number of designs and work within the same MOGA operators. Therefore, the only significant difference in algorithm performance comes from the starting population. However, a motivation for this work was to support the design by shopping paradigm. Since the idea in this paradigm is that a designer will update their preferred solution as they explore the space, it could be argued that more Pareto points provide a richer canvas. Table 7 lists the number of Pareto points for the two populations at the evaluation intervals listed in Fig. 11.

<table>
<thead>
<tr>
<th>Evaluations</th>
<th>When using random population</th>
<th>When using targeted population</th>
</tr>
</thead>
<tbody>
<tr>
<td>50,000</td>
<td>51</td>
<td>18</td>
</tr>
<tr>
<td>100,000</td>
<td>65</td>
<td>37</td>
</tr>
<tr>
<td>250,000</td>
<td>82</td>
<td>93</td>
</tr>
<tr>
<td>500,000</td>
<td>130</td>
<td>123</td>
</tr>
</tbody>
</table>

The results from Table 7 indicate that while the targeted population using one price point is able to dominate the random population solution, it often offers less Pareto points for the designer to explore. This outcome further motivates the desire to explore solution quality when multiple price points are used. The next sub-section investigates how the number of price points used to generate candidate designs influences the final solution.

C. Solutions using multiple price points

Although a single price point led to a frontier that dominated the solution of the randomly generated population, additional benefits can be realized by achieving a greater diversity of seed solutions for the targeted initial population. To first explore the effect of using multiple price points, the number considered is increased to seven evenly spaced points. Product lines created from the candidate designs also evenly represent the seven price points. Generating this targeted initial population required 3,070 evaluations.

Having previously shown that the one price point solution dominates the random population solution, the one price point solution now acts as the baseline for further comparison. Figure 12 depicts the progression of the two Pareto frontiers at 50,000, 100,000, 250,000 and 500,000 evaluations. At 50,000 evaluations, there is a noticeable improvement many of the Pareto points from the seven price point solution dominate the Pareto points from the one price point solution. As the number of evaluations increase, the two frontiers exhibit similar behavior in the upper-left region of the performance space (lower preference share, larger ACM). However, the seven price point solution strongly dominates the right-hand region of the performance space. For instance, a company exploring this solution space would now find a design that captures the same amount of ACM (such as 0.8) but captures 5% more preference share (75% instead of 70%).

The results from Fig. 12 indicate that increasing the number of price points used to generate the targeted population provides an advantageous outcome. This outcome is further supported by the data in Fig. 9, which showed that seven price points led to seven unique products for at least 20% of the respondents. However, when thirty price points were considered, the data in Fig. 10 was noticeable right-skewed. To explore if the algorithm performance advantage is maintained when a much larger number of price points is used, the number of price points considered is increased to thirty. Generating this targeted initial population required 8,456 evaluations.
Figure 12. Frontier progression using a targeted population with one price point and a targeted population with seven price points

Figure 13 shows the progression of the two Pareto frontiers when 7 and 30 price points are considered. Significantly, at 50,000 evaluations the 7 price point solution fully dominates the 30 price point solution. This suggests that the increased evaluations needed to generate the 30 price point targeted population may instead be more effectively used going through the MOGA operators of selection, crossover, and mutation. Further, by 500,000 evaluations the two frontiers are effectively on top of each other. This result, when combined with the information in Fig. 10, allows a conclusion to be drawn that excessively sampling the price structure to create candidate designs yields little to no improvement over a more moderate sampling.

In the previous section, it was shown the targeted population with one price point solution often had fewer Pareto points than the random population. When 7 and 30 price points are used, these solutions have noticeably more Pareto points than the random population solution after 100,000 evaluations. At 50,000 evaluations, however, these two solutions have less, as shown in Table 8. This is due to the initial overhead needed to create the candidate designs for the targeted population. Yet, at 500,000 evaluations, there is essentially no difference in the final number of Pareto points. This further supports the conclusion that excessive sampling of price is unnecessary.

<table>
<thead>
<tr>
<th>Evaluations</th>
<th>Random population</th>
<th>1 price point</th>
<th>7 price points</th>
<th>30 price points</th>
</tr>
</thead>
<tbody>
<tr>
<td>50,000</td>
<td>51</td>
<td>18</td>
<td>39</td>
<td>34</td>
</tr>
<tr>
<td>100,000</td>
<td>65</td>
<td>37</td>
<td>65</td>
<td>76</td>
</tr>
<tr>
<td>250,000</td>
<td>82</td>
<td>93</td>
<td>128</td>
<td>163</td>
</tr>
<tr>
<td>500,000</td>
<td>130</td>
<td>123</td>
<td>187</td>
<td>191</td>
</tr>
</tbody>
</table>

Table 8. Number of Pareto points identified
When considering solution quality and computational expense, the targeted population with 7 price points appears to be the most effective strategy for beginning the search. A comparison of the final solution at 500,000 evaluations between the randomly generated population and the targeted population using 7 price points is shown in Fig. 14. Here, the targeted population solution has greater solution diversity on both axes, and dominates a significant portion of the randomly generated population.
V. Conclusions and future work

As discussed in Section II, previous work by the authors had demonstrated that the use of targeted initial populations reduce the computational cost of a genetic search and improve solution quality for single objective market-based design problems. The original formulation for generating candidate designs for the targeted population involved finding the product configuration that maximized each respondent’s observed utility. At the market level, the goal was to maximize the share of preference that a product line would capture. This required a pricing structure for each attribute level to be defined outside of the optimization problem, inherently fixing the contribution margin of each product.

When competing business objectives are considered – such as preference share and a surrogate for profit – these price variables must be solved for within the optimization; adding increased design freedom and computational complexity. The goal of this paper was to demonstrate how the approach for creating candidate designs for a targeted initial population could be extended for such multiobjective optimization problems. By generating candidate designs at multiple price points, candidate designs are created that provide a richer representation of the solution space by accommodating both competing objectives. However, a design challenge arises in defining the number of price points to consider.

Results from an automobile product line feature packaging problem demonstrate that this richer sampling procedure does increase solution quality and reduce computational expense. Significant gains were seen along the share of preference axis, while limited returns were seen on the ACM axis. Further, the results from the case study problem suggest that the addition of too many price points provides limited returns. It was shown that the solution from a targeted population with 7 price points existed in the same performance regions as the targeted population with 30 price points. Additionally, while both of these solutions had more Pareto points than the solution using a randomly generated population – supporting the design by shopping paradigm – the difference in points between the 7 and 30 solutions were minimal. Therefore, while designers will see concrete benefits from adding multiple points when generating a targeted population, the results in this paper suggest a coarse sampling of the price axis may be the most effective strategy.

Extensions of this work will be pursued by exploring opportunities in three primary areas. First, the effects of varying genetic algorithm parameters such as mutation and crossover types/rates and size of the targeted population must be further explored. Performance metrics of interest will include Pareto domination, run-time, and solution diversity. Second, the case study results for the targeted population demonstrated a lack of solution uniformity, especially at high levels of ACM. For instance, at 500,000 evaluations, the solution generated from the random initial population appears to be more uniformly spaced across the performance space. While the targeted population identified more Pareto designs, there is also a “gap” in the frontier between 50 and 55% preference share. From a design by shopping perspective, a more uniform representation of the frontier may be desired. This uniformity may be achieved by exploring strategic approaches of creating the targeted population product lines from the pool of candidate designs. Finally, a third area of future work is the development of a run-time advisor capable of estimating minimum effective run-times, run-time to exhaustion, and potential payoffs of continued analysis. The combination of these efforts will support the extension of the targeted initial population approach to product family design. Here, schemas for platform commonality could be created by identifying trends within the pool of candidate designs for the targeted population.

Acknowledgments

We gratefully acknowledge support from the National Science Foundation through NSF Grant No. CMMI-0969961. Any opinions, findings, and conclusions presented in this paper are those of the authors and do not necessarily reflect the views of the National Science Foundation.

References


60Matlab, The MathWorks.