Conrad S. Tucker
Graduate Student
Mem. ASME
e-mail: ctucker4@uiuc.edu

Harrison M. Kim[1]
Assistant Professor
Mem. ASME
e-mail: hmkim@uiuc.edu

Department of Industrial and Enterprise Systems Engineering,
University of Illinois at Urbana-Champaign,
104 South Mathews Avenue,
Urbana, IL 61801

Optimal Product Portfolio Formulation by Merging Predictive Data Mining With Multilevel Optimization

This paper addresses two important fundamental areas in product family formulation that have recently begun to receive great attention. First is the incorporation of market demand that we address through a data mining approach where realistic customer preference data are translated into performance design targets. Second is product architecture reconfiguration that we model as a dynamic design entity. The dynamic approach to product architecture optimization differs from conventional static approaches in that a product architecture is not fixed at the initial stage of product design, but rather evolves with fluctuations in customer performance preferences. The benefits of direct customer input in product family design will be realized through the cell phone product family example presented in this work. An optimal family of cell phones is created with modularity decisions made analytically at the engineering level that maximize company profit. [DOI: 10.1115/1.2838336]

1 Introduction

The increased performance expectations of consumers and the volatility of today's leading market segments have forced companies to reevaluate their business models. The mass customization concept has become the revolutionary strategy for companies to better meet customer needs by shifting away from traditional product portfolios that satisfied only the average expectations of customers to more customer-specific product variants [1]. Mass customization, however, should be regulated so that customers do not become overwhelmed with an oversaturation of products to choose from [1]. The economic justifications of mass customization typically rely on the cost saving benefits of *economies of scale* that are due to the inventory reductions, uninterrupted manufacturing processes, etc. [2].

Commonality among product variants is a widely acceptable method of mitigating the inevitable cost increases of such highly differentiated products. By designing product variants around a shared and efficient product architecture, companies can reduce manufacturing and design costs associated with product differentiation [3]. The absence of standard performance metrics, however, has hindered consensus in this field in determining the best approach to solving this problem [4]. Under this product family design paradigm, we introduce a method to analytically determine the optimal product architecture configuration in the multiproduct hierarchy by directly incorporating what customers want (i.e., preference, performance expectations, etc.) in the design and formulation of a family of products. We propose an enterprise level objective that will serve as a generic model in applications dealing with product architecture design. The proposed enterprise profit function takes a new approach by linking customer performance preferences using data mining techniques, with engineering design capabilities in a dynamic setting. The term dynamic is used to describe the evolving product architecture that occurs due to the fluctuations in customer preference as attributes are included/excluded in the engineering model.

Currently, there exist module based and scale based methods of assessing product architecture design [4]. In this paper, the engineering design of the product architecture will be matched with the enterprise targets, acquired through realistic customer survey data, although the proposed framework is not limited to survey method. The data can be acquired from existing company databases. The primary focus is to present sufficient evidence of the profit maximization benefits that exist with the linking of performance targets in different product architectures, while still achieving desirable product performance. (Note that the focus is not how to collect the preference data, rather how to identify desired attributes in (large-scale) demand data, then link them with engineering design.)

We will observe in our cell phone example presented in this work that the benefits of sharing can be extracted directly from the predictive data mining model, with changes in attribute combinations and product architecture design. While module based product family design benefits can be directly observed by the manufacturing cost savings associated with modular architectures, we focus on observing the benefits of modularity through the fluctuations in customer purchasing price as modular components are selected/deselected among several architectures existing in the product family.

Multilevel, multidisciplinary optimization has become an effective alternative to solve complex, large-scale system design problems that are conventionally solved by all-in-one (AIO) approach. To link product design and product planning effectively, however, traditional static formulation should be expanded to dynamic formulations to model the changing product specifications and market demand. In this paper, individual product architectures are modeled using the hierarchical approach of analytical target setting (ATS) [5] and analytical target cascading (ATC) [6] and are further expanded to accommodate changing design variables and component sharing information. (Note that the proposed methodology is not limited to analytical target cascading, rather it can be combined with any generic multilevel optimization.)

The motivation of this research is to explain how multilevel design optimization can be extended in a multiproduct setting to

[1]Corresponding author.

Contributed by the Design Theory and Methodology Committee of ASME for publication in the Journal of Mechanical Design. Manuscript received October 31, 2006; final manuscript received October 22, 2007; published online March 20, 2008. Review conducted by Janet K. Allen. Paper presented at the 11th AIAA MAO Conference.

include the optimization of engineering designs for a product family in an extremely volatile and competitive market space.

This paper is organized as follows. This section provides a brief motivation and background. Section 2 describes previous works closely related to the current research. Section 3 describes the methodology. Section 4 demonstrates the methodology through a cell phone family design example. Section 5 presents the results and discussion. Section 6 concludes the paper.

2 Related Work

In recent years, significant interest has been paid to mass customization as it relates to product portfolio design as companies continue to become more customer specific. de Weck and Chang approach the product portfolio problem by allowing sales volume sensitivities and product variant performance to dictate the number of optimal product architectures [7]. Gonzales-Zugasti et al. used an *interactive implementation* approach that first establishes a product architecture design, then its variants [8]. This approach is engineering intensive with most of the product portfolio decisions made by engineers, rather than target customers. Other approaches by Desai et al. [9] and Kim and Chhajed [10] incorporate consumers into the product portfolio decisions and partition the consumer market into two groups; high-end and low-end customers, and design product variants based on the performance and quality expectations desired by each market.

Our approach gives the customer more control in the final outcome of a product by initially identifying a customer's maximum purchasing price, then allowing the engineering design level to select the quality and performance of the components included in that particular architecture. The predictive data mining approach that we incorporate in our methodology ensures that what customer wants are acquired directly from the customer, rather than being interpreted by enterprise decision makers. Along this line, Agard and Kusiak [11] employ data mining clustering techniques to segment a customer data set into candidate target markets for the design of product families. Association rule mining is then used to determine attribute patterns in the segmented data. Such data mining clustering techniques, however, still leave the enterprise decision maker with the daunting task of selecting the appropriate target market to pursue. Hence, decisions without initial engineering design validation may lead to an unsuccessful product portfolio. Data mining techniques are also investigated by Moon et al. [12] in identifying functional requirements to be applied to a predefined product architecture. Our approach to data mining fully utilizes its predictive capabilities by directly cascading customer wants to the engineering design of a product architecture. To allow for a more intelligent product architecture design, we opt to omit a predefined architecture, but rather start with an amorphous concept that quickly transforms into a customer ready product. For example, the outer design of a product is a function of the number of components present in it, some of which are modular and others, made to individual product specification. The final product architecture therefore will depend on customer specifications and engineering limitations. At this stage, product variants can then be manufactured based on this architecture, or if infeasible, another architecture will be introduced. The summation of *feasible* product variants will comprise the optimal product family. (The term feasible is used to signify an engineering design that can be both manufactured and at the same time completely meets customer expectations.)

3 Predictive Portfolio Design Methodology

The overall objective of this work is to establish an acceptable method of analytically designing a family of products that maximize the overall company profit while concurrently meeting performance expectations. In this section, core components of the predictive portfolio design methodology are described, where the product family configuration changes are allowed.

3.1 Predictive Modeling

3.1.1 Data Mining Approach. The acquisition of marketing data to determine patterns is vital to the overall stability and success of a company. Stored data can be related to manufacturing capabilities, consumer tendencies, distribution patterns, sales, etc. [13] The importance of properly analyzing data may be the distinguishing factor between success and failure. To this end, automated analysis and discovery tools that are powerful enough to analyze large data sets are becoming more popular.

In engineering product design and development, such powerful analysis can translate into lucrative project endeavors. A limiting factor in the manufacturing aspect of product design, however, is from the cost and functionality constraints placed by the pursued product market. A successful product portfolio requires that engineering capabilities are strongly matched with customer requirements. This can be a somewhat daunting task since traditional product design is a sequential process that starts with customer wants and linearly progresses until a final product is designed [13].

Our approach to product design hopes to alleviate some of the burdens of late stage design failures by making the customer-manufacturing relationship an iterative process, wherein a customer's preference is realized and updated with each manufacturing change in the desired architecture. To address the customer satisfaction aspect of product design, customer data can be acquired through a customer survey process and transformed into meaningful engineering design information (see Fig. 1).

By collecting customer data, data mining can determine relationships between inputs that were once unobservable [14]. There are several methods in which data mining tools can accomplish this task, but for the purpose of our research, we will focus on the Naïve Bayes approach in predicting a customer's maximum purchasing price (MaxPrice) that would yield the most efficient and profitable product portfolio. One should note that the concept of MaxPice is a time invariant metric that represents the customer's willingness to pay for a particular product at an instant in time. Cook proposes an S model to quantitatively determine the value added by the introduction of new product features [15]. Unlike conjoint analysis and similar methods [16], the data mining predictive approach employed in our work extracts previously *unknown* knowledge without requiring attribute ranking and complex matrix inverse operations by classifying attribute combinations based on the Naïve Bayes model expounded on in the next section.

3.1.2 Naïve Bayesian Model. The Naïve Bayes algorithm builds a predictive model based on supporting evidence from a fraction of the customer survey data, used to train the computer learning model [18]. Applied to a customer's maximum purchasing price, the Naïve Bayesian model can be posed as follows:

Given N elements in a set of customer attributes $a_i \epsilon A$. The dependent class variable MaxPrice(γ) has outcomes conditional on customer attributes $a_i,\ldots,a_N$ [17] $p(\gamma|a_1,\ldots,a_N) = p(\gamma)\cdot p(a_1,\ldots,a_N|\gamma)/p(a_1,\ldots,a_N) \rightarrow$ Probability of MaxPrice(γ), given certain input attribute(s) [18] Since the denominator of the above equation is independent of MaxPrice(γ) and the input attributes are known a priori, the denominator is essentially constant and can therefore be ignored [17].

Using the definition of conditional probability [19],

$$p(\gamma|a_1,\ldots,a_N) = p(\gamma)\cdot p(a_1,\ldots,a_N|\gamma) \tag{1}$$

$$= p(\gamma)\cdot p(a_1|\gamma)\cdot p(a_2,\ldots,a_N|\gamma,a_1) \tag{2}$$

$$= p(\gamma)\cdot p(a_1|\gamma)\cdot p(a_2|\gamma,a_1)\cdot p(a_3,\ldots,a_N|\gamma,a_1,a_2) \tag{3}$$

$$= p(\gamma)\cdot p(a_1|\gamma)\cdot p(a_2|\gamma,a_1)\cdot p(a_3|\gamma,a_1,a_2) \cdot p(a_4,\ldots,a_N|\gamma,a_1,a_2,a_3) \tag{4}$$

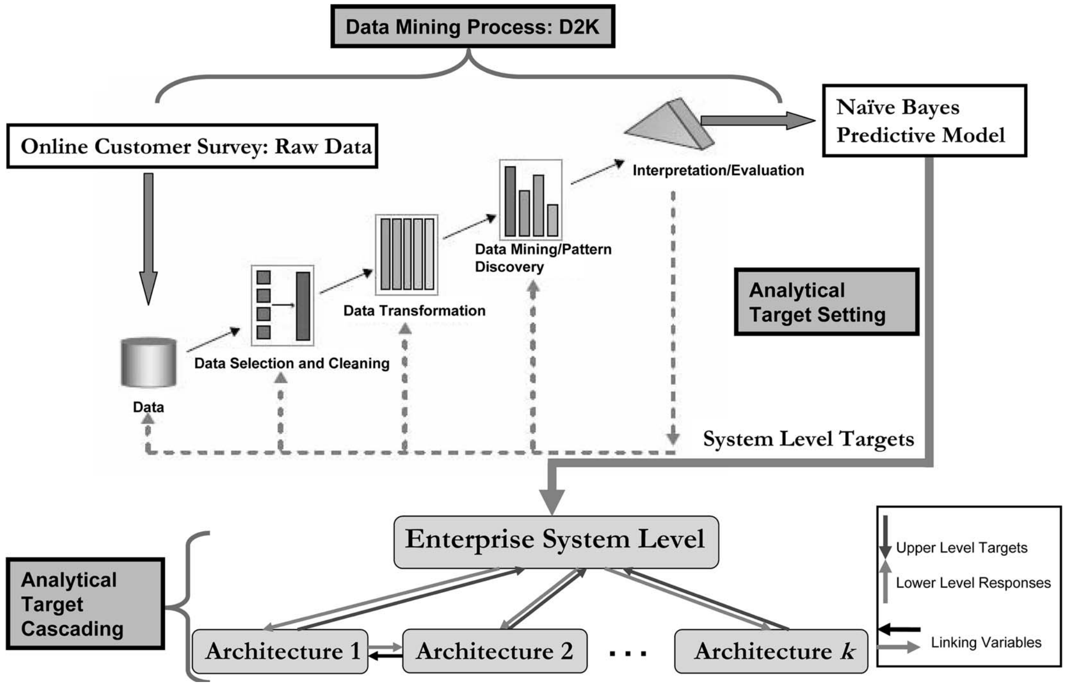

Fig. 1 Overall predictive product portfolio formulation (adapted from D2K manual [14])

The fundamental basis of the Naïve Bayesian model is the assumption of conditional independence of each input attribute, i.e., attribute a_i is independent of a_j where $i \neq j$ [17]. This is a valid assumption for our cell phone case study that will be expounded on later. For example, we make the assumption that the probability that a cell phone is a flip design, given a MaxPrice of \$200 is independent of the probability that a cell phone has a battery life of 5 h, given the same MaxPrice of \$200.

The assumption of independence enables us to express the conditional distribution of MaxPrice(γ) [20]

$$p(\gamma, a_1, \ldots, a_N) = p(\gamma) \prod_{l=k}^{n} p(a_i|\gamma) \tag{5}$$

A machine learning approach known as *Supervised Learning* [21] attempts to estimate the parameters of the developed Naïve Bayesian model. The assumption of attribute independence allows us to estimate the class variable (MaxPrice), prior to testing our model. The Naïve Bayes classifier combines the probability model with a decision rule; in most cases, a most probable hypothesis rule known as maximum a posteriori probability (MAP) [17] is computed, which determines the maximum likelihood of a given class. The function is modeled as follows [20]:

$$\text{classify}(a_1, \ldots, a_N) = \arg\max_{\gamma} p(\Gamma) \prod_{i=k}^{n} p(\Lambda_i|\Gamma) \tag{6}$$

where Γ takes on a value in the set γ, i.e., the value of MaxPrice must match a numerical value of one of the elements in the MaxPrice set.

Similarly, Λ_i takes on a value in the set a_i, i.e., attribute value i takes on a value of an element that exists in the overall attribute set, where arg max is the likelihood estimator of MaxPrice. Prior knowledge of the attribute distribution is assumed and a point estimate of the class variable can be obtained [22]. Based on the posterior distribution, the class variable γ is estimated as the statistical mode or in other words, the most recurring [22]. The Naïve Bayes classifier using the maximum a posteriori [20] decision rule is a valid approach in our study of customer predictive preferences, as the model takes into account a priori [23] preference of attributes. The robustness of the classifier validates the assumption of attribute independence and correctly predicts the class variable MaxPrice [17]. The following simple example illustrates the predictive strengths of the Naïve Bayes classifier in determining previously unknown knowledge from a given customer data set.

In the following example data set (Table 1), we have ten unique customer responses represented by each row. The three attribute types (Phone Type, Connectivity, Feature) are mutually exclusive and comprise of binary selections. For example, the first attribute Phone Type can assume one of two values, (Flip or Shell), etc. The objective is to determine what combination of attributes would result in a particular class variable prediction, i.e., purchase a phone (Yes or No). Let us assume that we are trying to classify a cell phone design that has the following attributes (Flip phone, Bluetooth, MP3). Note that this attribute combination *does not exist* in our example data set (Table 1) and such a classification would therefore be considered as new, previously unknown knowledge [24]. To determine the class (Purchase Phone=Yes or No) that such an attribute combination would fall under, we apply the conditional probability rule explained in Eq. (5). The conditional probabilities of each attribute are presented in Table 2 and the subsequent classification presented in the following calculations.

Table 1 Sample customer response data

	Attribute Selections			Class Variable
Customer	Phone Type	Connectivity	Feature	Purchase Phone
1	Flip	Wifi	MP3	NO
2	Shell	Bluetooth	Camera	YES
3	Shell	Wifi	MP3	NO
4	Shell	Bluetooth	MP3	NO
5	Flip	Wifi	Camera	NO
6	Flip	Wifi	MP3	YES
7	Shell	Bluetooth	MP3	YES
8	Flip	Bluetooth	Camera	NO
9	Flip	Wifi	MP3	YES
10	Flip	Wifi	Camera	NO

Table 2 Conditional probability calculations for each attribute

Conditional Probabilities	Class Prediction: Purchase Phone=YES (4 occurrences)	Phone=NO (6 occurrences)
P(Phone Type=Flip ∣ Purchase Phone=YES)	2/4	
P(Connectivity=Bluetooth ∣ Purchase Phone=YES)	2/4	
P(Feature=MP3 ∣ Purchase Phone=YES)	3/4	
P(Phone Type=Flip ∣ Purchase Phone=NO)		4/6
P(Connectivity=Bluetooth ∣ Purchase Phone=NO)		2/6
P(Feature=MP3 ∣ Purchase Phone=NO)		3/6

$$\begin{aligned} &\mathrm{p(YES|Flip\ Phone, Bluetooth, MP3)} \\ &\quad = \mathrm{p(YES)} \cdot \mathrm{p(Flip\ Phone|YES)} \\ &\qquad \cdot \mathrm{p(Bluetooth|YES)} \cdot \mathrm{p(MP3|YES)} \\ &\quad = \frac{4}{10}\frac{2}{4}\frac{2}{4}\frac{3}{4} = 0.075 \end{aligned} \tag{7}$$

$$\begin{aligned} &\mathrm{p(NO|Flip\ Phone, Bluetooth, MP3)} \\ &\quad = \mathrm{p(NO)} \cdot \mathrm{p(Flip\ Phone|NO)} \\ &\qquad \cdot \mathrm{p(Bluetooth|NO)} \cdot \mathrm{p(MP3|NO)} \\ &\quad = \frac{6}{10}\frac{4}{6}\frac{2}{6}\frac{3}{6} = 0.067 \end{aligned} \tag{8}$$

The maximum likelihood function utilized by the Naïve Bayes model selects the class variable with the maximum likelihood of occurring, which in this case would be Purchase Phone=YES with a probability of 0.075. In other words, this *new* combination of cell phone attributes has the potential of appealing to the consumer market and would therefore be a candidate cell phone design. Such powerful insights have the potential to significantly enhance the product family formulation process as attribute combinations can be analyzed and optimized to achieve a more efficient product development strategy.

This example is a simplified version of the actual customer preference data utilized in this work, which comprises of a customer data set of 100,000 and a wider array of attributes. Despite such a large data set, the final Naïve Bayes predictive results took less than 300 s to generate running on an Intel Pentium IV desktop (3.2 GHz).

3.1.3 Data Mining Using Data to Knowledge (D2K). The term Knowledge Discovery in Databases (KDD) describes the entire process of extracting data from large-scale databases [14]. The process begins with the acquisition of realistic customer preference data through a comprehensive online survey that is posed to capture the product performance expectations of customers [25]. The results of this survey will be used by an innovative data mining tool know as *Data to Knowledge* (D2K) that classifies the results and maps (classifies) the data into one of several predefined classes [14]. The process from data extraction to predictive model is as follows (Fig. 1).

Step 1: Data Acquisition: Importing the raw data from a database (e.g., *SQL Server*). Customer preference data can be acquired in several ways: In many instances, customer preference data exist in large databases and is known to the enterprise decision maker through customer behavior tracking methods [24] (credit card purchases, coupons promotions, etc). Another approach to acquiring such data is through an interactive online survey. In our example a web survey was designed and created using webtools interactive software[2], wherein users could automatically fill out and submit the survey results. The webtools software is configured so as to automatically save the results of the web survey in a CSV file format in EXCEL, which can be then directly extracted and used for analysis purposes such as data mining supervised machine learning. Each row in the survey raw data results represents a submission by a particular unique customer/individual, with the associated attributes stored in the corresponding columns.

The fundamental strength of data mining as opposed to other customer survey analysis tools such as conjoint analysis is the ability to analyze large data sets in an efficient manner. To reveal these strengths, the initial customer response data are extrapolated to simulate the response behavior of a data set of 100,000 customers. This raw data set of 100,000 customers is used in our data mining analysis and subsequent product demand predictions. It is important to note that the demand prediction is an instant in time. Time varying stochastic behavior of demand and price are topics for later works.

Step 2: Data Selection and Cleaning: The stage where irrelevant or noisy data are identified and removed and relevant data are extracted from the raw data [26].

Step 3: Data Transformation: This phase involves the transformation of data into acceptable forms for the data mining process. Here, irrelevant attributes are eliminated to improve the model's predictive power. Data transformations can include binning, normalizing, missing value imputation, outlier removal, etc. [27].

Step 4: Data Mining/Pattern Discovery: First, a particular algorithm is selected and for the predictive analysis for our cell phone architecture design, we have opted to use a Naïve Bayes model.

Step 5: Interpretation and Evaluation: Typically, D2K uses 2/3 of the raw data to train the machine and the remaining 1/3 to test the model developed. The N-fold cross validation technique selects and compares several test models with one another and selects the appropriate model that best predicts the class variable [14].

The predictive model will enable a seamless translation of customer data into tangible design targets for the engineering design level. Selection or deselection of attributes to observe the effects on the class variable (MaxPrice) can be formulated as a *mixed integer programming* problem [28], where our objective is to search through a combination of attributes that would yield the MaxPrice and market share percentages needed to maximize the overall profit of the company.

The visual representation in Fig. 2 is the D2K graphical user interface output that enables the user to manually select/deselect attributes that influence the MaxPrice prediction. The square box enclosing each attribute indicates which attributes are active in the predictive model. Only one parameter value per attribute can be active at once due to the Naïve Bayes assumption of attribute

[2]www.webtools.uiuc.edu

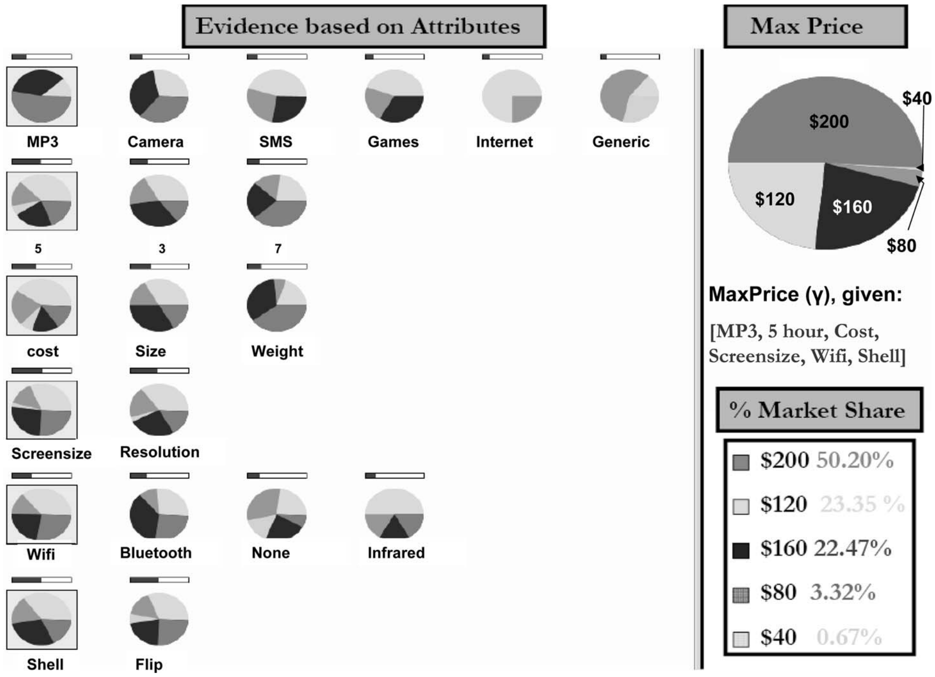

Fig. 2 D2K Naïve Bayes prediction of maximum customer purchasing price (MaxPrice) and associated market share α_i

independence. The Naïve Bayesian prediction of MaxPrice has a percentage value associated with each MaxPrice prediction, which translates into the percentage of customers with the same MaxPrice prediction. If none of the attributes is selected, then the MaxPrice prediction is calculated solely on the initial state of information, i.e., the surveyed customers and their overall preferences.

The active attributes in the prediction of MaxPrice are set as targets at the engineering design sublevel, while the MaxPrice is used to determine the enterprise profit for product variant i. In order to ensure an optimal product that satisfies customer wants, customer targets cascaded down to the engineering sublevel are weighted more than any other objective in the engineering sublevel, such as cost minimization.

The iterative process of trying to match customer targets with the engineering capabilities yields an optimal product that is both profitable and is desired by customers. The overall process from D2K's Naïve Bayes prediction of MaxPrice to product design and development using multilevel optimization (analytical target setting [5] and analytical target cascading [29] are utilized in our approach) in the engineering level is illustrated in Fig. 1.

Commonality is achieved by the linking of component variables among architectures. Our methodology suggests that commonality decisions be made on the primary basis of how they affect customer preferences and ultimately, enterprise profit. The Mcconnell/Stigler relationship between unit cost and output suggests that diseconomies of scale may mitigate the cost-savings benefit that commonality provides to the manufacturing process as output increases exponentially [30,31]. Therefore, the benefits of commonality and modularity will focus less on the manufacturing cost savings, but rather on overall company profit. The reason for this performance metric shift is due to the ambiguities that exist when product manufacturing cost is the primary reason for justifying sharing decisions. Such cost minimization commonality decisions may have adverse effects on the satisfaction of intended customers who may suffer due to the performance sacrifices in an attempt to reduce cost. Future research aspirations include incorporating the entire supply chain process into the product family cost model to better understand the effects of downstream processes in enterprise decision making.

3.2 Product Portfolio Optimization at Product Family Supersystem Level. The primary product portfolio objective of launching product architectures is achieved through a finite launch of product architectures deemed most profitable by the enterprise system level objective. The profit maximizing objective is realized through an iterative process of acquiring the MaxPrice a customer is willing to pay for a particular product (determined by a customer predictive model), and the cost derived from the component selection process that defines that particular product. The overall maximum profit (π_{overall}) is used as the metric for this selection process, where profit$_{\text{overall}}$ is the summation of the individual product profits that would yield the maximum overall company profit. The product portfolio limit used in our case study is assumed to be the maximum number of product variants in the manufacturing process that would allow the process to still remain efficient, i.e., the point of inflection before capacity and distribution capabilities are unmanageable by a company.

The flow diagram in Fig. 3 illustrates the iterative process of product portfolio development and the product family mathematical model is summarized as follows.

Minimize

$$-\sum_{j=1}^{k} x_j \cdot \pi_{\text{variant}(j)} \tag{9}$$

where $\pi_{\text{variant}(j)}$ is the profit of variant (j), x_j the binary discrete variable selecting or deselecting particular product variant (π_{variant_j}) where $\sum_{j=1}^{k} x_j \leq K$, k the total feasible product architectures that can be designed, K the product portfolio limit (number of architectures in the product family),

Subject to

$$h1: \quad x_j = \{0,1\} \quad f \in \{1, \ldots, k\} \tag{10}$$

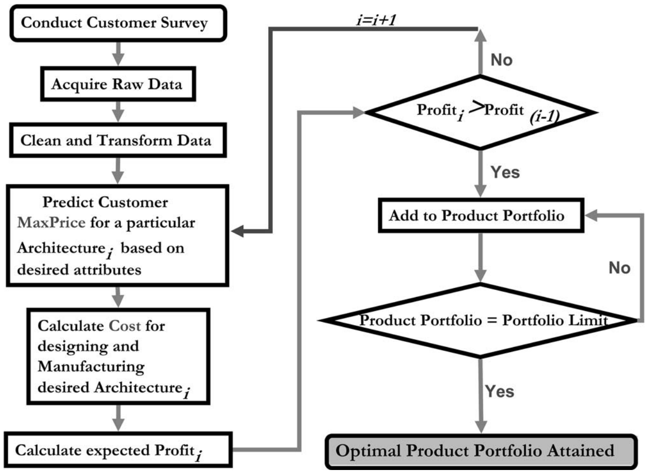

Fig. 3 Data flow of product portfolio formulation

$$g1: \quad \sum_{j=1}^{k} x_j - K \leqslant 0 \tag{11}$$

Product portfolio limit K is a finite number meaning that a company cannot produce every possible combination of architectures, which would be an impractical real life target.

3.3 Enterprise System Level. The Naïve Bayes predictive model allows for a customer's MaxPrice value to be used in determining the maximum variant$_j$ profit (π_{variant_j}), where profit$_j$ for a particular product is determined by

$$\pi_{\text{variant}_{\{j\}}} = \text{MaxPrice}_{\{j\}} - \text{cost}_j \tag{12}$$

where MaxPrice$_j$ is the Naïve Bayesian prediction based on certain input attributes. The MaxPrice can be partitioned during the customer survey to N number of price preference choices to reflect the objective of the enterprise decision maker. $i=[1,\ldots,N]$. The *most probable* (as defined by Eq. (6), class variable is used in profit calculation in Eq. (12)) cost$_j$ is the engineering sublevel response for the cost needed to produce a product desired by the customers, based on the Naïve Bayes prediction.

The mathematical model at the enterprise level is summarized as follows. (The norm notation indicates $\|\cdot\| = \|\cdot\|_2^2$, i.e., squared L-2 norm.)

Minimize

$$-\pi_{\text{variant}_j} + \|T^C - R^{\text{ent}}\| + \varepsilon_R + \varepsilon_y \tag{13}$$

Subject to

$$h1: \quad \pi_{\text{variant}} - D \cdot \text{MaxPrice}_{\{j\}} - \text{cost}_{\{j\}} = 0 \tag{14}$$

$$h2: \quad \sum_{i=1}^{m}\sum_{j=1}^{n} a_{i,j} - m = 0 \tag{15}$$

$$h3: \quad D - \text{market demand}_{\text{variant}_j} = 0 \tag{16}$$

$$h4: \quad \sum_{i=1}^{N} \alpha_i - 1 = 0 \tag{17}$$

$$g1: \quad \|R^{\text{eng}} - R^{\text{eng}^L}\| - \varepsilon_R \leqslant 0 \tag{18}$$

$$g2: \quad \|y - y^L\| - \varepsilon_y \leqslant 0 \tag{19}$$

($h2$: Given an $m \times n$ matrix of attributes, equality constraint $h2$ restricts the parameter value of each attribute to only one per row due to the Naïve Bayesian assumption of attribute independence.) where π_{variant_j} is the profit of product variant j; T^C the product variant target component predicted by Naïve Bayes customer model; R^{ent} the engineering response component cascaded up to the enterprise level; $R^{\text{ent}}=R^{\text{ent}}(x^{\text{ent}},R^{\text{eng}})$, meaning that the enterprise level response is a function of system variables and the response of the engineering subsystem level; y the linking variables at the enterprise level, the linking variable concept applied to product family design represent shared attributes or components that exist among product variants; y^L the target values for linking variable at the engineering subsystem level cascaded up to enterprise level; ε_R the deviation tolerance between customer component targets and engineering response; ε_y the deviation tolerance between linking variables.

3.4 Engineering Design Subsystem Level. The engineering design subsystem level is defined as the stage in product design wherein engineering design objectives and constraints are formulated to produce a product/variant that satisfies the enterprise level objective [6].

3.4.1 Analytical Target Setting [5]. The multiobjective formulation of the engineering design sublevel focuses primarily on designing an architecture (around which product variants are designed) at product launch that will satisfy customer wants, predicted by the Naïve Bayesian model, while simultaneously minimizing the overall cost of the product. Meeting what customer wants is weighted more due to the obvious reasons; a cheaper product will not automatically translate into an attractive product if customer wants are not satisfied. The mathematical model at the engineering design level is summarized as follows:

Minimize

$$\text{cost}_j + \|R^{\text{eng}^U} - R^{\text{eng}}\| + \|y_j - y_j^U\| \tag{20}$$

where cost$_j$ is the cost of product variant (j); R^{eng^U} the response from the enterprise system level, cascaded down to the engineering level (at the enterprise system level mathematical formulation, R^{eng^U} is simply R^{eng}); R^{eng} the response from the engineering sublevel, i.e., $R^{\text{eng}}=R^{\text{eng}}(x_{\text{eng}},y_{\text{eng}})$, meaning that the response of the engineering design sublevel is a function of local design variables and also sharing linking variables (R^{eng} at the engineering subsystem level will become R^{eng^L} at the enterprise system level); y_j the linking variables at the engineering design level; y_j^U the target value for linking variables from the upper enterprise level cascaded down to engineering level,

Subject to

$$h1: \quad \text{cost} - \sum_{i=1}^{J} x_i q_i = 0 \tag{21}$$

$$\text{design constraints:} \quad g_j(x_{\text{eng}}) \leqslant 0 \tag{22}$$

where x_i is the binary discrete variable selecting or deselecting particular product variant component (q_{variant}); q_i the product variant component (discrete or continuous variable): *discrete component variable* is purchased from a manufacturer with predefined performance and cost attributes and *continuous component variable* is company manufactured with changing specifications to cater to dynamic architecture design; J the total available components in the engineering product design.

The engineering design subsystem level objective is modeled as a mixed integer nonlinear programming problem with discrete variables that dictate the component selection process and continuous variables for the engineering designed components. A branch and bound algorithm is used to achieve an optimal solution [28]. Since there are both discrete and continuous variables in our mathematical model, the branch and bound algorithm attempts to find an optimal solution by first solving the relaxation problem (i.e., integer restrictions are relaxed), which is simply a nonlinear optimization problem [28]. In the subsequent solution, if all the

Table 3 Customer survey questions and response options

Survey questions	Survey answer choices
What feature would you most like your cellular phone to have?	MP3, camera, internet, games, SMSText, just talk
What is more important to you?	Weight, size, or cost of cell phone
What type of cell phone design do you like?	Flip phone design, shell phone design
What type of connectivity would you prefer your phone to have?	Bluetooh, WiFi, infrared, none
What is the minimum talk time you require before a recharge?	3 hours, 5 hours, 7 hours
What is more important to you?	Display screen size, display resolution
What is the maximum price you would be willing to pay for the features you just described	\$40, \$80, \$120, 160, \$200

discrete variables take integer values, then the mixed integer problem is solved and an optimal solution is reached [28]. For each discrete variable that does not take on an integer value, the algorithm takes this variable and divides the problem (branches) into two new nonlinear programming problems. This process is continued until a global optimum is achieved.

4 Application

4.1 Product Portfolio Formulation: Cellular Phone Product Family. To demonstrate the effectiveness of our approach, we will apply the proposed methodology to a realistic cell phone market to determine the optimal family of cellular architectures that would satisfy a captured market demand. We begin by introducing a customer survey questionnaire (Table 3) that is modeled to realistically capture the true essence of what the customer wants, acquired through realistic customer survey data, although the proposed framework is not limited to survey method. The data can be acquired from existing company databases. Performance metrics determined by the customer prediction will be set as targets at the engineering level.

From the results of the survey, a model can now be developed that will predict the maximum purchasing price that a customer is willing to pay based on certain attributes. The D2K software helps to develop this model with the transformation of the customer raw data. MaxPrice can then be used in a sensitivity analysis to determine the profit for a particular product design, given certain selected attributes. The attributes selected are used as targets in the engineering level.

4.2 Enterprise System Level

4.2.1 Cell Phone System Profit Optimization. The Naïve Bayesian model developed by D2K allows the user to select/deselect attributes and observe the change in MaxPrice and the market share associated with each. As can be seen in Fig. 2, a MP3 phone architecture is used as a starting point with customer attributes including (5 h Battery life [32], Cost Objective, Screen Size Priority, Wifi Connectivity [33], and Shell Phone Design).

These attribute targets are then cascaded down to the engineering sublevel to determine whether or not such a product design is feasible. The MaxPrice prediction is used at the enterprise level in calculating the profit(π) for this particular MP3 phone. Mathematically, this is represented as

Minimize

$$-\pi_{\text{MP3 variant}_1}+\|T^{\text{battery life}}-R^{\text{battery life}^{\text{ent}}}\|+\|T^{\text{Wifi}^{\text{ent}}}-R^{\text{Wifi}^{\text{ent}}}\|+\|T^{\text{shell}}-R^{\text{shell}^{\text{ent}}}\|+\varepsilon_{\text{battery life}}+\varepsilon_{\text{Wifi}}+\varepsilon_{\text{shell}} \quad (23)$$

In the cell phone case study, $R^{\text{battery life}^{\text{ent}}}$ is considered as a linking variable at the engineering design level. Thus, a deviation constraint $g1$ is added in the constraint set.

Subject to

$$h1: \quad \pi_{\text{MP3 variant}_1}-D(\text{MaxPrice}_{\{\text{MP3}_1\}}-\text{cost}_{\{\text{MP3}_1\}})=0 \quad (24)$$

$$h2: \quad \text{MaxPrice}_{\{\text{MP3}_1\}}-\$200=0 \quad (25)$$

$$h3: \quad \sum_{i=1}^{m}\sum_{j=1}^{n}a_{i,j}-m=0 \quad (26)$$

$$h4: \quad \sum_{i=1}^{K}\alpha_i-100\%=0 \quad (27)$$

$$h5: \quad \alpha_i\equiv\{51\%,23\%,22\%,3\%,1\%\} \quad (28)$$

$$h6: \quad \text{MaxPrice}_i=\{200,120,160,80,40\} \quad (29)$$

$$g1: \quad \|R^{\text{battery life}^{\text{ent}}}-R^{\text{battery life}^{\text{eng}^L}}\|-\varepsilon_{\text{battery life}}\leqslant 0 \quad (30)$$

$$g2: \quad \|R^{\text{Wifi}^{\text{ent}}}-R^{\text{Wifi}^{\text{eng}^L}}\|-\varepsilon_{\text{Wifi}}\leqslant 0 \quad (31)$$

$$g3: \quad \|R^{\text{shell}^{\text{ent}}}-R^{\text{shell}^{\text{eng}^L}}\|-\varepsilon_{\text{shell}}\leqslant 0 \quad (32)$$

$$g4: \quad D-D_0\leqslant 0 \quad (33)$$

where $\pi_{\text{MP3variant}_j}$ is the profit (in \$) of MP3 variant with specific design features; $T^{\text{battery life}}$ the battery life (hours) target predicted by customer Naïve Bayes model; $R^{\text{battery life}}$ the battery life response cascaded up from engineering sublevel; T^{Wifi} the connectivity (bluetooth, wifi, or infrared) target predicted by customer Naïve Bayes model; R^{Wifi} the connectivity response cascaded up from engineering sublevel; T^{shell} the shell design target predicted by customer Naïve Bayes model; R^{shell} the shell design response cascaded up from engineering sublevel; y the linking variables at the enterprise level, the linking variable concept applied to product family design represent shared attributes or components that exist among product variants. y the target values for linking variable at the engineering subsystem level cascaded up to enterprise level; $\varepsilon_{\text{battery life}}$ the deviation tolerance between customer component targets and engineering response; $\varepsilon_{\text{Wifi}}$ the deviation tolerance between customer component targets and engineering response; $\varepsilon_{\text{shell}}$ the deviation tolerance between customer component targets and engineering response.

Here, $D_0=100{,}000$ (represents the total market population of cell phone consumers) and K is 5 (product Portfolio limit that would enable manufacturing process to remain efficient). The table of demand information for a given class variable prediction is given in Table 4, where $D=8395$ represents the demand for a \$200 phone based on the Naïve Bayesian model in Eq. (6). One of the values of MaxPrice will be selected for each attribute combination (customer predicted preference) so long as it is *more probable* than any other class variable of MaxPrice (see Eq. (6)).

4.3 Engineering Subsystem Level

4.3.1 Cell Phone Subsystem. The engineering subsystem level comprises of a multiobjective function of cost minimization while simultaneously minimizing the deviation between customer de-

Table 4 Demand information based on the Naïve Bayesiann predictive model

MaxPrice	Customer demand (D) at given price
\$200	8395
\$160	16,001
\$120	21,796
\$80	12,908
\$40	7899

sign targets and engineering response. During the first iteration in the product portfolio optimization, linking variables are nonexistent due to the fact that only one optimal cell phone design exists in the product portfolio set.

The basic mathematical formulation of successive cell phone variants is similar to perturbations occurring with each successive product variant (Fig. 4), depending on the customer targets.

Minimize

$$\mathrm{cost}_{\mathrm{MP3\ variant}_1} + \|R^{\mathrm{battery\ life}^U} - P^{\mathrm{battery\ life}}\| + \|R^{\mathrm{Wifi}^U} - R^{\mathrm{Wifi}}\| + \|R^{\mathrm{shell}^U} - R^{\mathrm{shell}}\| \tag{34}$$

Subject to the following:

In screen resolution constraints,

$$h1:\quad (A1 * \mathrm{LCD}_{\mathrm{length}} * \mathrm{LCD}_{\mathrm{width}}) - \mathrm{LCD}_{\mathrm{res}} = 0 \tag{35}$$

$$h2:\quad (A2 * \mathrm{LCD}_{\mathrm{length}} * \mathrm{LCD}_{\mathrm{width}}) - \mathrm{cost}_{\mathrm{LCD}} = 0 \tag{36}$$

$$h3:\quad (A3 * \mathrm{LCD}_{\mathrm{length}} * \mathrm{LCD}_{\mathrm{width}}) - \mathrm{weight}_{\mathrm{LCD}} = 0 \tag{37}$$

$$h4:\quad (A4 * \mathrm{LCD}_{\mathrm{length}} * \mathrm{LCD}_{\mathrm{width}}) - \mathrm{power}_{\mathrm{LCD}} = 0 \tag{38}$$

$$h5:\quad (A5 * \mathrm{OLED}_{\mathrm{length}} * \mathrm{OLED}_{\mathrm{width}}) - \mathrm{OLED}_{\mathrm{res}} = 0 \tag{39}$$

$$h6:\quad (A6 * \mathrm{OLED}_{\mathrm{length}} * \mathrm{OLED}_{\mathrm{width}}) - \mathrm{cost}_{\mathrm{OLED}} = 0 \tag{40}$$

$$h7:\quad (A7 * \mathrm{OLED}_{\mathrm{length}} * \mathrm{OLED}_{\mathrm{width}}) - \mathrm{weight}_{\mathrm{OLED}} = 0 \tag{41}$$

$$h8:\quad (A8 * \mathrm{OLED}_{\mathrm{length}} * \mathrm{OLED}_{\mathrm{width}}) - \mathrm{power}_{\mathrm{OLED}} = 0 \tag{42}$$

In battery design constraints,

$$h9:\quad \mathrm{cap}_{\mathrm{NIMH}} - (\mathrm{NIMH}_{\mathrm{const1}} * (\mathrm{V}_{\mathrm{NIMH}})) - T_{\mathrm{Hours}} * \sum_{i=1}^{N} P_{\mathrm{component}_i} = 0 \tag{43}$$

$$h10:\quad \mathrm{cap}_{\mathrm{LION}} - (\mathrm{LION}_{\mathrm{const1}} * (V_{\mathrm{LION}})) - T_{\mathrm{hours}} * \sum_{i=1}^{N} P_{\mathrm{component}_i} = 0 \tag{44}$$

$$h11:\quad \mathrm{battery}_{\mathrm{talk\ time}} - (\mathrm{NIMH} * ((0.0053 * (\mathrm{capacity}_{\mathrm{NIMH}})) + 0.0269) + (\mathrm{LION} * ((0.0061 * (\mathrm{capacity}_{\mathrm{LION}})) + 0.1667))) = 0 \tag{45}$$

$$h12:\quad ((\mathrm{NIMH}_{\mathrm{const2}} * (L_{\mathrm{NIMH}} * W_{\mathrm{NIMH}} * T_{\mathrm{NIMH}})) - \mathrm{cost}_{\mathrm{NIMH}}) = 0 \tag{46}$$

$$h13:\quad ((\mathrm{LION}_{\mathrm{const2}} * (L_{\mathrm{LION}} * W_{\mathrm{LION}} * T_{\mathrm{LION}})) - \mathrm{cost}_{\mathrm{LION}}) = 0 \tag{47}$$

$$h14:\quad ((\mathrm{NIMH}_{\mathrm{const3}} * (L_{\mathrm{NIMH}} * W_{\mathrm{NIMH}} * T_{\mathrm{NIMH}})) - Wg_{\mathrm{NIMH}}) = 0 \tag{48}$$

$$h15:\quad ((\mathrm{LION}_{\mathrm{const3}} * (L_{\mathrm{LION}} * W_{\mathrm{LION}} * T_{\mathrm{LION}})) - Wg_{\mathrm{LION}}) = 0 \tag{49}$$

$$g1:\quad (\mathrm{NIMH} * L_{\mathrm{NIMH}} + \mathrm{LION} * L_{\mathrm{LION}}) - 0.60 * (\mathrm{shell} * L_{\mathrm{shell}} + \mathrm{flip} * L_{\mathrm{shell}}) \leqslant 0 \tag{50}$$

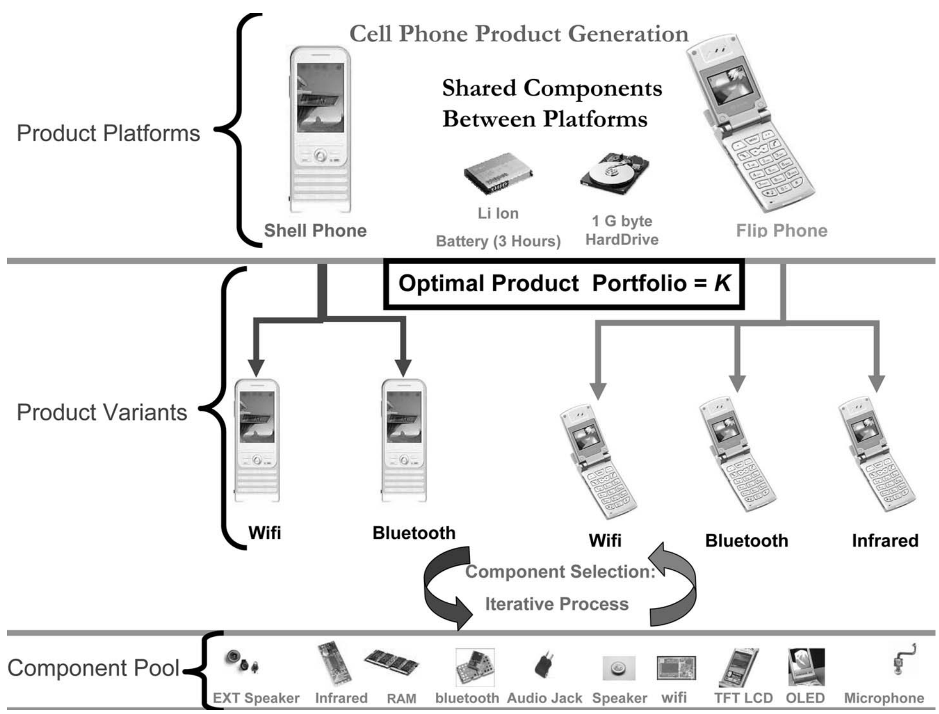

Fig. 4 Optimal product portfolio example. Illustrates how just two product architectures can generate product variants that make up a family of products (product portfolio of *K* =5 products).

Table 5 Possible shared component variables

Component	Description	Design Options
Internal memory (RAM)	32 Mbytes RAM discrete choice variable	Manufacturer
Internal memory (RAM)	64 Mbytes RAM discrete choice variable	Manufacturer
External memory	Memory stick pro discrete choice variable	Manufacturer
External memory	Memory stick duo discrete choice variable	Manufacturer
Hard drive	1 Gbytes storage discrete choice variable	Manufacturer
Hard drive	2 Gbytes storage discrete choice variable	Manufacturer
Phone design	Shell phone design variables	Engineeringdesign
Phone design	Flip phone design variables	Engineeringdesign
Battery type	Lithium polymer [34] battery design variables	Engineeringdesign
Battery type	Lithium ion [34] battery design variables	Engineeringdesign
Connectivity	Bluetooth connection discrete variable	Manufacturer
Connectivity	Wifi discrete choice variable	Manufacturer
Connectivity	Infra red discrete choice variable	Manufacturer
Auto codec	Microphone discrete variable	Manufacturer
Auto codec	Earpiece discrete variable	Manufacturer
Auto codec	Auto jack discrete variable	Manufacturer
Auto codec	External speaker discrete variable	Manufacturer
Display type	TFT LCD [35] discrete variable	Manufacturer
Display type	OLED [35] discrete variable	Manufacturer

$$g2: \quad (\mathrm{NIMH} * W_{\mathrm{NIMH}} + \mathrm{LION} * W_{\mathrm{LION}}) - 0.95 * (\mathrm{shell} * W_{\mathrm{shell}} + \mathrm{flip} * W_{\mathrm{flip}}) \leqslant 0 \tag{51}$$

$$g3: \quad (\mathrm{NIMH} * T_{\mathrm{NIMH}} + \mathrm{LION} * T_{\mathrm{LION}}) - 0.45 * (\mathrm{shell} * T_{\mathrm{shell}} + \mathrm{flip} * T_{\mathrm{flip}}) \leqslant 0 \tag{52}$$

(To enhance the overall flow of the paper, several variable names are abbreviated (L=Length, W=Width, T=Thickness, Wg=Weight, V=Volume, Cap=Capacity, P=Power consumption, etc.))

In cell phone outer casing design constraints

$$h16: \quad (\mathrm{shell}_{\mathrm{const1}} * L_{\mathrm{shell}} * W_{\mathrm{shell}} * T_{\mathrm{shell}}) - \mathrm{cost}_{\mathrm{shell}} = 0 \tag{53}$$

$$h17: \quad (\mathrm{flip}_{\mathrm{const1}} * L_{\mathrm{flip}} * W_{\mathrm{flip}} * T_{\mathrm{flip}}) - \mathrm{cost}_{\mathrm{flip}} = 0 \tag{54}$$

$$h18: \quad (\mathrm{shell}_{\mathrm{const2}} * L_{\mathrm{shell}} * W_{\mathrm{shell}} * T_{\mathrm{shell}}) - Wg_{\mathrm{shell}} = 0 \tag{55}$$

$$h19: \quad (\mathrm{flip}_{\mathrm{const2}} * L_{\mathrm{flip}} * W_{\mathrm{flip}} * T_{\mathrm{flip}}) - Wg_{\mathrm{flip}} = 0 \tag{56}$$

$$g4: \quad L_{\mathrm{LCD}} - (0.60 * \mathrm{shell} * L_{\mathrm{shell}} + 0.60 * \mathrm{flip} * L_{\mathrm{flip}}) \leqslant 0 \tag{57}$$

$$g5: \quad (0.30 * \mathrm{shell} * L_{\mathrm{shell}} + 0.30 * \mathrm{flip} * L_{\mathrm{flip}}) - L_{\mathrm{LCD}} \leqslant 0 \tag{58}$$

$$g6: \quad w_{\mathrm{LCD}} - 0.90 * (\mathrm{shell} * W_{\mathrm{shell}} + \mathrm{flip} * W_{\mathrm{flip}}) \leqslant 0 \tag{59}$$

$$g7: \quad 0.7 * (\mathrm{shell} * W_{\mathrm{shell}} + \mathrm{flip} * W_{\mathrm{flip}}) - W_{\mathrm{LCD}} \leqslant 0 \tag{60}$$

$$g8: \quad L_{\mathrm{OLED}} - (0.60 * \mathrm{shell} * L_{\mathrm{shell}} + 0.60 * \mathrm{flip} * L_{\mathrm{flip}}) \leqslant 0 \tag{61}$$

$$g9: \quad (0.30 * \mathrm{shell} * L_{\mathrm{shell}} + 0.30 * \mathrm{flip} * L_{\mathrm{flip}}) - L_{\mathrm{OLED}} \leqslant 0 \tag{62}$$

$$g10: \quad W_{\mathrm{OLED}} - 0.90 * (\mathrm{shell} * W_{\mathrm{shell}} + \mathrm{flip} * W_{\mathrm{flip}}) \leqslant 0 \tag{63}$$

$$g11: \quad 0.7 * (\mathrm{shell} * \mathrm{width}_{\mathrm{shell}} + \mathrm{flip} * \mathrm{width}_{\mathrm{flip}}) - \mathrm{OLED}_{\mathrm{width}} \leqslant 0 \tag{64}$$

In design objectives,

$$h20: \quad \text{total cost} - \sum_{i=1}^{N} \text{component}(i)_{\mathrm{cost}} = 0 \tag{65}$$

$$h21: \quad \text{total weight} - \sum_{i=1}^{N} \text{component}(i)_{\mathrm{weight}} = 0 \tag{66}$$

Table 5 identifies the possible shared components of each individual MP3 capable phone. Sharing decision are influenced by customer performance expectations and engineering capabilities. Certain components are purchased directly from a manufacturer and would therefore have fixed performance specifications, while other components can be manufactured by the company to meet customer needs.

4.4 Optimization Study. With a methodological approach to product architecture formulation, enterprise decision makers can have a validation tool to justify product portfolio formulation and launch decisions. We begin by predefining a finite number of unique architectures that will constitute our optimal product family. For our study, a product portfolio of five product variants will be set as our maximum manufacturing ability. It is assumed in this case study that a product portfolio greater than five will begin to result in diseconomies of scale and ultimately, reduced profit [30].

Predictive product performance targets are acquired through our data mining process and used to set our initial starting point values at the enterprise level as targets for the engineering subsystem level, i.e., (MP3, 5 h Battery Life, Cost Objective, Screensize, Wifi Connectivity, Shell Design). For this particular cell phone design, we get an engineering product design cost of \$98.3/unit. The cost response from the engineering design level is then cascaded up to the enterprise level where the MaxPrice is used to calculate the predicted profit (π_{variant}). With a predicted MaxPrice of \$200 and an associated demand (D) of \$8395, we arrive at a projected profit of \$853,448. The particular product design, however, fails to meet the battery life target of 5 h, instead designing a cell phone with a battery life of only 4.5 h.

The customer focused objective of matching predicted performance expectations and the engineering design objective of designing the lowest cost product are competing objectives. The enterprise profit calculations presented in this work are the projected profit calculations for a given product launch, based on a particular phone design and how closely it matches customer performance expectations (which may consequently affect the product demand). We make the argument that it is better to launch a product that has lower projected profits (but fully satisfies customer expectations) than to launch a product with a higher projected profit margin (but fails to satisfy customer wants). Failure to satisfy customer wants would adversely affect the *actual* demand for that product and decrease the actual enterprise profit as customers switch to alternative products that more closely satisfy their performance expectations. The disparity between *projected* versus *actual* profit calculations is therefore highly dependent on product performance.

4.4.1 Optimal Product Portfolio Model. Mathematically, this optimization process is translated to

Minimize

$$\sum_{j=1}^{k} -x_j\left(D_j \cdot \left(\pi_{\text{variant}(j)} - \lambda \cdot \sum_{i=0}^{N} \varepsilon_i\right)\right) \tag{67}$$

$\pi_{\text{variant}(j)}$ is the profit for variant (j); x_j the binary discrete variable selecting or deselecting particular product variant ($\pi_{\text{variant}(j)}$) where $\Sigma_{j=1}^{k} x_j \leq K$, where K is the optimal product portfolio limit of 5 for our example and k symbolizes the nine studied MP3 architectures; D_j, the demand for a given product design λ, the weighted value for penalty term ε_i; ε_i, the tolerance deviation term for particular customer target (i); N, the total number of shared components.

The comparative analysis can now begin where we use the above calculated profit ($853,448) as our base. We will first determine the profits of the first five product variants that can be feasibly designed with the MP3 technology. For each successive iteration, we will compare the newly calculated profit of variant_j to that of each $\text{variant}_{i[i=1,\ldots,K]}$ existing in our feasible product family set. If the newly calculated product variant profit is greater than any of the variant profits in our set, we will discard the least profitable and replace it with variant_j.

Depending on the number of possible combinations of the predictive model, either an exhaustive search approach or a tree branching algorithm can be used. For the MP3 Architecture, nine combinations are analyzed with a battery life of 3 h as our primary sharing component.

5 Results and Discussion

To determine our optimal product portfolio of five architectures, the nine combinations of MP3 capable cell phones were analyzed (Table 6) to determine the profit margins of each product variant. The optimization results reveal that variant_1 fails to satisfy customer targets on one of the performance metrics, i.e., deviation between customer battery life target and engineering battery manufacturing capabilities $\|T^{\text{battery life}} - R^{\text{battery life}}\|$ is greater than tolerance $\varepsilon_{\text{battery life}}$ and is therefore deemed less profitable with the incorporation of the penalty term described in Eq. (67).

Each subsequent product variant uses a battery life of 3 h, which may initially seem less profitable due to the changes in the Naïve Bayes predictions of MaxPrice (Figs. 2 and 4). After an engineering design validation, however, we see that such architectures would be cheaper to manufacture and would yield the highest profit margins while satisfying customer wants. (Note: An engineering response is an evaluation of technical capabilities by the engineering team in determining the feasibility of such a product. The relevance to battery life is that certain product concepts may have unattainable battery life expectations.)

The optimal product portfolio (Fig. 4) given this approach will therefore be architectures {3,5,6,7,9}, yielding a total company profit of

$$\begin{aligned}\pi_{\text{overall}} &= \$848,902 + \$881,596 + \$967,996 + \$1,024,232\\ &\quad + \$930,769 = \$4.65\ \text{Million}\end{aligned} \tag{68}$$

The multilevel optimization solution (adopting the ATC methodology [17]) took approximately 500 s per product variant running on an Intel Pentium IV desktop (3.2 GHz). The model was developed in the Matlab [36] environment with TOMLAB [31] used in the optimization sequence.

The cost-savings benefit of manufacturing can be realized when a product manufacturing process has minimal number of interruptions. Thus the more components that a product shares with variants, the higher the probability that this may translate into lower overall company operating costs. Sharing decisions focused solely on manufacturing process cost savings can, however, have adverse effects on customer preferences and ultimately their willingness to pay as seen in the following example. Four out of the nine product architectures share a flip phone design (Tables 6 and 7). Although it would be more desirable for all architectures to share the same type of design (flip or shell), it is clearly observed that such a decision would not yield the most profitable product portfolio. For example, sharing a shell phone design for 5 architectures would mean selecting architectures {1,2,4,6,8}, which would yield a maximum profit of

$$\begin{aligned}\pi_{\text{overall}} &= \$853,448 + \$774,642 + \$807,336 + \$967,996\\ &\quad + \$632,093 = \$4.03\ \text{Million}\end{aligned} \tag{69}$$

Even without penalizing variant_1 for failing to satisfy the customer battery target of 5 h (*actual engineering response*=4.5 h), we see that a sharing decision of a shell phone design would yield a less profitable product portfolio. The solution to product portfolio optimization is multifaceted requiring input from different specializations across different boundaries. Such powerful insights will help enterprise decision makers understand the intricate link that exists between what customer wants and engineering design capabilities.

6 Conclusion

The emergence of a customer driven need for product differentiation has lead companies to re-evaluate current design and manufacturing processes [37]. Consequently, analytical techniques are required to alleviate the costs associated with product differentiation. The greatest challenge is to develop an optimal product architecture for a family of products in a dynamic market space. To overcome this challenge, we have successfully demonstrated how data mining techniques can help analytically develop a product family by encompassing customer requirements directly with engineering capabilities using ATS [5] and ATC [6]. Modularity and component sharing decisions can now be expanded beyond manufacturing cost savings to include consumer price sensitivity to product architecture changes. The dynamic product architecture concept utilized in this work has the benefit of continuously changing architecture design variables throughout the product design phase to cater to customer preference requirements. A product portfolio is achieved, which not only maximizes profit but simultaneously satisfies what the customer wants. The validity of this method enables us to expand and include multistage problems, especially focusing on a changing market space.

The cell phone analysis systematically attains a feasible product portfolio by simultaneously focusing on changing demand due to a particular product design choice. The model places emphasis on deterministic (and in later works stochastic) methods in product architecture formulation. The long term goal is to provide decision makers in industry with a useful tool that helps mitigate the associated risks involved in product portfolio formation and product launch decisions. Such a tool has the potential to drastically reduce errors associated with ad hoc product portfolio methodologies or disjointed expertise between the business and engineering teams. The manufacturing benefits of product architecture design and product portfolio formulation will be incorporated in later models to reflect a wider scope of product design. Careful attention will be paid to the efficiency at which the algorithm of choice will converge to an optimal solution. An exhaustive search algo-

Table 6 Optimal Product Family Results (Part 1). Highlighted architectures represent the most profitable product portfolio given maximum five architectures allowed in the portfolio.

Variable description	Component source	MP3 Phone1 solution	MP3 Phone2 solution	MP3 Phone3 solution	MP3 Phone4 solution	MP3 Phone5 solution	MP3 Phone6 solution	MP3 Phone7 solution	MP3 Phone8 solution	MP3 Phone9 solution	Units
32 Mbytes discrete variable	Manufacturer 1	0	1	1	1	1	1	1	1	1	—
	Manufacturer 2	0	0	0	0	0	0	0	0	0	—
	Manufacturer 3	0	0	0	0	0	0	0	0	0	—
64 Mbytes discrete variable	Manufacturer 1	0	0	0	0	0	0	0	0	0	—
	Manufacturer 2	1	0	0	0	0	0	0	0	0	—
	Manufacturer 3	0	0	0	0	0	0	0	0	0	—
Memory stick pro discrete variable	Manufacturer 1	0	0	0	0	0	0	0	0	0	—
	Manufacturer 2	0	0	0	0	0	0	0	0	0	—
	Manufacturer 3	0	0	0	0	0	0	0	0	0	—
Memory stick duo discrete variable	Manufacturer 1	0	0	0	0	0	0	0	0	0	—
	Manufacturer 2	0	0	0	0	0	0	0	0	0	—
	Manufacturer 3	0	0	0	0	0	0	0	0	0	—
1 Mbytes storage discrete variable	Manufacturer 1	0	0	0	0	0	0	0	0	0	—
	Manufacturer 2	1	0	0	0	0	0	0	0	0	—
	Manufacturer 3	0	0	0	0	0	0	0	0	0	—
2 Mbytes storage discrete variable	Manufacturer 1	0	0	0	0	0	0	0	0	0	—
	Manufacturer 2	0	0	0	0	0	0	0	0	0	—
	Manufacturer 3	0	0	0	0	0	0	0	0	0	—
Shell phone discrete variable	Engineering design	1	1	0	1	0	1	0	1	0	—
Phone length	Engineering design	85.0	80.0	120.0	80.0	120.0	80.0	120.0	80.0	120.0	mm
Phone width	Engineering design	48.4	40.0	40.0	40.0	40.0	40.0	40.0	40.0	40.0	mm
Phone thickness	Engineering design	19.1	17.7	16.3	17.7	16.3	16.5	12.0	17.4	12.0	mm
Phone weight	Engineering design	40.0	28.9	40.0	28.9	40.0	27.0	29.4	28.4	29.4	g
Phone cost	Engineering design	18.0	13.0	18.0	13.0	18.0	12.1	13.2	12.8	13.2	$
Flip phone discrete variable	Engineering design	0	0	1	0	1	0	1	0	1	—
Phone length	Engineering design	100.0	100.0	100.0	100.0	100	0.135.1	100.0	100.0	100.0	mm
Phone width	Engineering design	45.0	68.0	45.0	68.0	45.0	45.0	45.0	45.0	45.0	mm
Phone thickness	Engineering design	18.1	12.0	12.0	12.0	12.0	12.0	12.0	18.1	12.0	mm

Table 6 (*Continued.*)

Variable description	Component source	MP3 Phone1 solution	MP3 Phone2 solution	MP3 Phone3 solution	MP3 Phone4 solution	MP3 Phone5 solution	MP3 Phone6 solution	MP3 Phone7 solution	MP3 Phone8 solution	MP3 Phone9 solution	Units
Phone weight	Engineering design	40.0	40.0	28.5	40.0	26.5	35.7	26.5	40.0	26.5	g
Phone cost	Engineering design	12.0	12.0	7.9	12.0	7.9	10.7	7.9	12.0	7.9	s
Nickel metal hydride battery discrete variable	Engineering design	0	0	0	0	0	0	0	0	0	—
Battery weight	Engineering design	42.4	49.2	50.0	50.0	50.0	50.0	50.0	50.0	50.0	g
Battery length	Engineering design	46.2	51.1	36.6	59.7	72.5	38.2	80.0	70.0	80.0	mm
Battery width	Engineering design	54.7	34.2	48.6	37.0	32.3	60.0	60.0	50.4	25.5	mm
Battery thickness	Engineering design	17.9	30.0	30.0	24.1	22.7	23.3	11.1	15.1	26.1	mm
Battery cost	Engineering design	17.1	19.9	20.2	20.2	20.2	20.2	20.2	20.2	20.2	$
Battery capacity	Engineering design	803.9	958.9	962.6	976.8	962.6	1016.8	1002.6	986.8	972.6	mAh
Lithium ion battery discrete variable	Engineering design	1	1	1	1	1	1	1	1	1	$
Battery weight	Engineering design	17.76	12.82	13.11	12.82	13.11	12.00	12.29	12.62	12.91	g
Battery length	Engineering design	51.00	48.00	72.00	48.00	72.00	48.00	72.00	48.00	63.58	mm
Battery width	Engineering design	45.97	38.00	42.75	38.00	42.75	38.00	42.75	38.00	42.74	mm
Battery thickness	Engineering design	8.58	7.96	4.52	7.96	4.82	7.45	4.52	7.83	5.38	mm
Battery cost	Engineering design	16.16	11.67	11.93	11.67	11.93	10.92	11.19	11.48	11.74	$
Battery capacity	Engineering design	706.77	464.47	464.47	464.47	464.47	464.47	464.47	464.47	464.47	mAh
Cell phone talk time	Engineering design	4.48	3.00	3.00	3.00	3.00	3.00	3.00	3.00	3.00	h
Bluethooth discrete variable	Manufacturer 1	0	0	0	1	1	0	0	0	0	—
WIFI discrete variable	Manufacturer 1	1	1	1	0	0	0	0	0	0	—
Intre red discrete variable	Manufacturer 1	0	0	0	0	0	0	0	1	1	—

Table 7 Optimal Product Family Results (Part 2). Highlighted architectures represent the most profitable product portfolio given maximum five architectures allowed in the portfolio.

Variable description	Component Source	MP3 Phone1 Solution	MP3 Phone2 Solution	MP3 Phone3 Solution	MP3 Phone4 Solution	MP3 Phone5 Solution	MP3 Phone6 Solution	MP3 Phone7 Solution	MP3 Phone8 Solution	MP3 Phone9 Solution	Units
Microphone discrete variable	Manufacturer 1	1	1	1	1	1	1	1	1	1	—
	Manufacturer 2	0	0	0	0	0	0	0	0	0	—
EarPiece discrete variable	Manufacturer 1	0	0	1	0	1	0	1	0	1	—
	Manufacturer 2	0	0	0	0	0	0	0	0	0	—
	Manufacturer 3	1	1	0	1	0	1	0	1	0	—
Audio jack discrete variable	Manufacturer 1	0	0	0	0	0	0	0	0	0	—
	Manufacturer 2	0	0	0	0	0	0	0	0	0	—
External speaker discrete variable	Manufacturer 1	0	0	0	0	0	0	0	0	0	—
	Manufacturer 2	1	0	0	0	0	0	0	0	0	—
	Manufacturer 3	1	1	1	1	1	1	1	1	1	—
LCD discrete variable	Engineering design	1	1	1	1	1	1	1	1	1	—
Length of LCD discrete variable	Engineering design	25.50	24.00	30.00	24.00	30.00	24.00	30.00	24.00	30.00	mm
Width of LCD	Engineering design	34.94	26.00	31.50	28.00	31.50	28.00	31.50	28.00	31.50	mm
Display resolution	Engineering design	13131.32	8905.28	13929.30	9905.28	13929.30	9905.28	13929.30	9905.28	13929.30	pixels
LCD manufacturing cost	Engineering design	4.45	3.36	4.73	3.38	4.73	3.36	4.73	3.36	4.73	$
LCD unit weight	Engineering design	35.63	26.88	37.60	26.88	37.870	26.88	37.80	26.88	37.80	g
LCD power consumption	Engineering design	6.91	6.72	9.45	6.72	9.45	6.72	9.45	6.72	9.45	mAh
OLED discrete variable	Engineering design	0	0	0	0	0	0	0	0	0	—
Length of OLED	Engineering design	25.50	35.33	30.00	25.57	30.00	33.41	31.75	33.21	30.00	mm
Width of OLED	Engineering design	39.22	28.09	31.50	36.00	31.50	28.00	31.50	28.00	33.33	mm
Display resolution	Engineering design	19620.00	19473.58	18540.90	18050.00	18540.90	18352.50	19620.00	18245.00	19520.00	pixels
OLED manufacturing cost	Engineering design	8.00	7.94	7.56	7.38	7.56	7.48	8.00	7.44	8.00	$
OLED unit weight	Engineering design	30.00	29.78	28.35	27.61	28.35	28.06	30.00	27.90	30.00	g

Table 7 (*Continued.*)

Variable description	Component Source	MP3 Phone1 Solution	MP3 Phone2 Solution	MP3 Phone3 Solution	MP3 Phone4 Solution	MP3 Phone5 Solution	MP3 Phone6 Solution	MP3 Phone7 Solution	MP3 Phone8 Solution	MP3 Phone9 Solution	Units
OLED power consumption	Engineering design	30.00	29.78	28.35	27.61	28.35	26.06	30.00	27.90	30.00	MhA
1 MegaPixel camera discrete variable	Manufacturer 1	0	0	0	0	0	0	0	0	0	—
	Manufacturer 2	0	0	0	0	0	0	0	0	0	—
2 MegaPixel camera discrete variable	Manufacturer 1	0	0	0	0	0	0	0	0	0	—
	Manufacturer 2	0	0	0	0	0	0	0	0	0	—
MP3 module discrete variable	Manufacturer 1	0	0	0	0	0	0	0	0	0	—
	Manufacturer 2	0	1	1	1	1	1	1	1	1	—
	Manufacturer 3	0	0	0	0	0	0	0	0	0	—
	Manufacturer 4	0	0	0	0	0	0	0	0	0	—
Internet module discrete variable	Manufacturer 1	0	0	0	0	0	0	0	0	0	—
	Manufacturer 2	0	0	0	0	0	0	0	0	0	—
Graphics module for games: discrete variable	Manufacturer 1	0	0	0	0	0	0	0	0	0	—
	Manufacturer 2	0	0	0	0	0	0	0	0	0	—
	Manufacturer 3	0	0	0	0	0	0	0	0	0	—
SMS text message technology discrete variable	Manufacturer 1	0	0	0	0	0	0	0	0	0	—
Total architecture cost	Engineering solution	98.34	84.46	81.05	82.96	79.58	75.68	73.04	80.60	73.80	$
Total architecture weight	Engineering solution	141.82	121.29	129.69	117.39	125.79	105.83	116.07	116.33	125.19	g

rithm or a branch and bound algorithm will help to ensure a global optimum for the maximum attainable profit for a family of architectures.

Acknowledgment

The authors acknowledge the support from the Automated Learning Group at the National Center for Supercomputing Applications. This material is based upon the work supported by the National Science Foundation under Award No. 0726934. Any options, findings, and conclusions or recommendations expressed in this publication are those of the authors and do not necessarily reflect the views of the National Science Foundation.

Nomenclature

K = product portfolio limit (maximum number of existing products at launch)

T^C = variant target component predicted by Naïve Bayes customer model

R^E = engineering design response (feasible/infeasible)

y = linking variable at the engineering sub-system level cascaded up to enterprise level

Λ_i = attribute selection (can assume a range of values)

γ = the maximum price (MaxPrice) a customer is willing to pay for a particular product design

π = projected profit of a feasible product design based on engineering design and predicted demand

ε_R = deviation tolerance between customer performance targets and engineering response

ε_y = deviation tolerance between linking variables

References

[1] Jiao, J., and Zhang, Y., 2005, "Product Portfolio Identification Based on Association Rule Mining," Comput.-Aided Des., **37**(2), pp. 149–172.

[2] Hsu, H., and Wang, W., 2000, "Dynamic Programming for Delayed Product Differentiation," Eur. J. Oper. Res., **156**, pp. 183–193.

[3] Simpson, T. W., 2004, "Product Platform Design and Customization: Status and Promise," Artif. Intell. Eng. Des. Anal. Manuf., **18**, pp. 3–20.

[4] Scott, M. J., Arenillas, J., Simpson, T. W., Valliyappan, S., and Allada, V., 2006, "Towards a Suite of Problems for Comparison of Product Platform Design Methods: A Proposed Classification," *Proceedings of DETC 06, 2006 ASME Design Engineering Technical Conferences*, Philadelphia, PA, Paper No. DETC2006/DAC-99289.

[5] Cooper, A. B., Georgiopoulos, P., Kim, H. M., and Papalambros, P. Y., 2006, "Analytical Target Setting: An Enterprise Context in Optimal Product Design," ASME J. Mech. Des., **128**(1), pp. 4–13.

[6] Kim, H. M., Michelena, N. F., Papalambros, P. Y., and Jiang, T., 2003, "Target Cascading in Optimal System Design," ASME J. Mech. Des., **125**(3), pp. 474–480.

[7] de Weck, O., Suh, E., and Chang, D., 2003, "Product Family and Platform Portfolio Optimization," *Proceedings of DETC03, 2003 ASME Design Engineering Technical Conferences*, Chicago, Il, Paper No. DETC03/DAC-48721.

[8] Gonzales-Zugasti, J. P., Otto, K. N., and Baker, J. D., 2001, "Assessing Value in Platformed Product Family Design," Res. Eng. Des., **13**(1), pp. 30–41.

[9] Desai, P., Kekre, S., Radhakrishnan, S., and Srinivasan, K., 2001, "Product Differentiation and Commonalityin Design: Balancing Revenue and Cost Drivers," Manage. Sci., **47**(1), pp. 37–51.

[10] Kim, K., and Chhajed, D., 2000, "Commonality in Product Design: Cost Saving, Valuation Change and Cannibalization," Eur. J. Oper. Res., **125**(3), pp. 602–621.

[11] Agard, B., and Kusiak, A., 2004, "Data Mining Based Methodology for the Design of Product Families," Int. J. Prod. Res., **42**(15), pp. 2955–2969.

[12] Moon, S. K., Kumara, S. R. T., and Simpson, T. W., 2006, "Data Mining and Fuzzy Clustering to Support Product Family Design," *Proceedings of DETC 06, 2006 ASME Design Engineering Technical Conferences*, Philadelphia, PA, Paper No. DETC2006/DAC-99287.

[13] 2002, *Data Mining for Design and Manufacturing*, Dan Braha, ed., Springer.

[14] McEntire, J., 2003, *D2K Toolkit User Manual*, National Center for Supercomputing Applications (NCSA).

[15] Cook, H. E., 1997, *Product Management: Value, Quality, Cost, Price, Profit, and Organization*, Chapman and Hall, London.

[16] Dean, D. H., 2004, "Evaluating Potential Brand Associations Through Conjoint Analysis and Market Simulation," J. Product Brand Management, **13**(7), pp. 506–513.

[17] Rish, I., 2001, "An Empirical Study of the Naive Bayes Classifier," *IJCAI 2001 Workshop on Empirical Methods in Artificial Intelligence*.

[18] Zhang, H., Ling, C., and Zhao, Z., 2000, "The Learnability of Naive Bayes," *Canadian AI2000, LNAI 1822*, pp. 432–441.

[19] Meretakis, D., and Wuthrich, B., 1999, "Extending Naive Bayes Classifiers Using Long Itemsets," *Fifth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 164–174.

[20] Flack, P. A., and Lachiche, N., 2004, "Naïve Bayes Classification of Structured Data," Mach. Learn., **57**(3), pp. 233–269.

[21] Kleinberg, E. M., 2000, *Lecture Notes in Computer Science*, Springer, Berlin, Vol. 1857.

[22] Degroot, M., 1970, *Optimal Statistical Decisions*, McGraw-Hill, New York.

[23] Sparacino, G., Tombolato, C., and Cobelli, C., 2000, "Maximum-Likelihood Versus Maximum a Posteriori Parameter Estimation of Physiological System Models: The c-Peptide Impulse Response Case Study," IEEE Trans. Biomed. Eng., **47**(6), pp. 801–811.

[24] Fayyad, U. and Uthurusamy, R., 2007, "Data Mining and Knowledge Discovery in Databases," Communications of the ACM, **39**(11), p. 24(3).

[25] Tucker, C., and Kim, H. M., 2006, "Cell Phone Customer Survey: Online Interactive User Interface Created Using Webtools," https://webtools.uiuc.edu/survey/Secure?id=5617516.

[26] Boulicaut, J., Esposito, F., Giannotti, F., and Pedreschi, D., 2004, *Knowledge Discovery in Databases: PKDD 2004*, Springer, New York.

[27] Campos, M., Stengard, P., and Milenova, B., 2005, "Data-Centric Automated Data Mining," *Fourth International Conference on Machine Learning and Applications*.

[28] Holmstrom, K., Goran, A. O., and Edvall, M. M., 2006, *Users Guide for TOMLAB/MINLP*, Tomlab Optimization, Tomlab Optimization Inc.

[29] Kim, H. M., Rideout, D. G., Papalambros, P. Y., and Stein, J. L., 2003, "Analytical Target Cascading in Automotive Vehicle Design," ASME J. Mech. Des., **125**(3), pp. 481–489.

[30] Canback, D., 2003, "Diseconomies of Scale in Large Corporations," Technical description, Canback Dangel Predictive Analytics Advisors, 10 Derne Street, Boston, MA 02114.

[31] Milliken and Company, 2006, "Cost Savings Benefit of Manufacturing," private communication.

[32] Cell Phone Battery Warehouse. Standby and talk times. 2006, http://www.batteries4less.com/.

[33] Neuvo, Y. 2004, "Cellular Phones as Embedded Systems," *IEEE International Solid-State Circuits Conference*.

[34] Buchmann, I., 1999, "Battery Mystery Solved: Why Batteries for Digital Cell Phones Fail," *Batteries Conference on Applications and Advances*, pp. 359–362.

[35] Klepper, M., Miller, P., and Miller, L., 2003, "Advanced Display Technologies, Printing Industry Center at Rochester Institute of Technology (RIT).

[36] 1999, The Math Works, Inc., Natick, MA, *Optimization Toolbox for Use with MATLAB*, version 2.

[37] Berry, S., and Pakes, A., 2007, "The Pure Characteristics Demand Model," International Economic Review, **48**(4), pp. 1193–1225