A Computationally Assisted Methodology for Preference-Guided Conceptual Design

Garrett J. Barnum
Research Assistant
e-mail: garrett@burley.com

Christopher A. Mattson[1]
Assistant Professor
e-mail: mattson@byu.edu

Department of Mechanical Engineering,
Brigham Young University,
Provo, UT 84602

We present an interactive, computationally assisted, methodology for capturing and incorporating designer preferences into a numerical search for design concepts. An initial pool of manually created designs is parameterized and used in a computational search that recombines features to form new designs in a semi-automated way. Designs are evaluated quantitatively by performance calculations and evaluated qualitatively by human designers. Designer preference is captured when visual representations of designs are presented to the designer for subjective evaluation. The methodology searches for optimally performing designs, guided by quantitative performance models and designer preferences. The methodology couples the speed of computational searches with the ability of human designers to subjectively evaluate unmodeled objectives. The new methodology is demonstrated with a vehicle architecture example, which generates a set of designs that progressively improves in performance and more fully meets designer preference. The proposed method brings the ability to generate numerous, optimally performing solutions across a wide solution space, in an efficient and human-centered way, and does so in the early stages of design. [DOI: 10.1115/1.4002838]

1 Introduction

Among the most important decisions in the product development process is one that marks the end of the conceptual design phase: the selection of the most promising design concept, which will be fully developed in the remaining phases of the development process. While there are various effective methodologies to assist the designer in identifying the best performing concepts within a given set [1–5], these methods are limited by the quality and quantity of the *set of concepts* under consideration. The quality of that set is partially determined by the level of creativity, intuition, and experience of the design team while the quantity is primarily determined by the amount of time and effort given to concept generation activities. Unfortunately, the abstract, ambiguous, and open-ended nature of conceptual design makes it impractical to generate and consider all, or even a majority of, the possible concepts using manual methods.

In early phases of design, there are often subjective decisions made, which are difficult to justify or explain rationally. It would be helpful to be able to model these subjective decisions since they can have a large impact on the final performance of a design. Traditional *physics-based models*, parametric models used to approximate or predict the physical performance of a design, can be described with equations, calculated using design variables, and numerically optimized. However, the existence of subjective decisions implies that there are important aspects of performance that are unmodeled by physics-based models but *are* modeled in the mind and intuition of the experienced designer, and that these subjective decisions are influential during conceptual design. Completely excluding humans from an automated evaluation process would ignore the importance of human subjectivity in decisions made during the conceptual design phase.

The purpose of the methodology presented in this paper is to form *preference-based models*, quantitative models of the qualitative preference of a designer, and incorporate them into a numerical search for better performing, more preferred designs. The methodology is intended to be used as a decision-making assist/tool during conceptual design, after an initial set of manually generated concepts has been generated, but before the selection of a final concept. The design methodology automates the traditionally manual process of combining and recombining features from an initial set of concepts, and incorporates the interactively formed preference-based models along with existing physics-based models, in order to guide the search for better performing combinations of the human-generated ideas/features. We note that the methodology presented here does not account for combinatorial effects or unequal weighting of designer's preference. This is the subject of our future paper.

The remaining parts of this paper are organized as follows. Section 2 provides a background on conceptual design automation and preference capture methods. Section 3 presents the proposed computationally assisted methodology. The methodology uses and builds on the automated, computational search methods developed in previous works [6,7] by the authors. The methodology will be applied to a vehicle architecture example in Sec. 4. This paper concludes and discusses future research in Sec. 5.

2 Literature Survey

In order to efficiently search through the vast number of possible solutions, conceptual design automation research [8–14] utilizes computational power to quickly search for and evaluate alternatives. Even in simple design problems, the number of variables and number of possible combinations of features can be beyond the capacity of human designers to manage manually and understand the impact of design changes to the overall design performance. To handle the large amounts of data, software programs have been developed that use digital morphological charts [8], allowing designers to visually select options from the chart. Design repositories are a way of digitally archiving solutions from existing products based on the functions they perform. The stored information is searched to find similar solutions that are then suggested for a new product with the same functions [9–11]. Currently, researchers at Missouri University of Science and Technology in collaboration with the University of Texas at Austin have produced a concept generator that draws on a design repository

[1]Corresponding author.

Contributed by the Design Theory and Methodology Committee of ASME for publication in the Journal of Mechanical Design. Manuscript received June 28, 2010; final manuscript received October 13, 2010; published online November 23, 2010. Assoc. Editor Janet K. Allen.

with stored design information from over 100 consumer products to produce alternative design solutions [12,13]. Other research by Hutcheson et al. showed that genetic algorithms can be used to select multiple designs for detailed evaluation based on quantitative objectives formulated during conceptual design [14]. Other work has shown the use of pattern search algorithms for efficient two- and three-dimensional packaging, layout, and routing problems [15,16]. These methods aid in combinatorially forming new designs in an automated way but the formation process does not account for any unmodeled objective that can be identified using the experience of a human designer.

Our previous works [6,7] presented a numerical optimization search strategy for automatically exploring morphological charts using genetic algorithm optimization search methods. The search strategy begins with an initial pool of creatively produced design concepts, which are decomposed in a general manner, grouping subfunction solutions/features into rows of a morphological chart. A design is numerically represented by a *design chromosome* c, defined as $c=[x_1 \quad x_2 \quad \dots \quad x_n]^T$, where x is a set of n number of design variables/features from the morphological chart. In the method of the previous works, genetic algorithms are used to automatically generate new designs by recombining values of the design chromosomes to find the best performing designs according to calculated objective values. It was demonstrated that the method could computationally search the vast number of possible combinations of features in a morphological chart and find optimal, nondominated, designs with increasing performance from generation to generation in the genetic algorithm. Simplified visual representations of the final set of designs and the calculated objective values were presented to designers for evaluation, enabling them to better understand and make trade-offs between the quantitative and qualitative aspects of the designs. While these methods form designs automatically and use traditional numerical optimization to find better performing designs quickly, if human subjectivity and preference could be included from the criteria of the automated search and evaluation process, the process would more closely match the criteria used by human designers in traditional conceptual design.

Design experience and knowledge are very valuable and difficult to transfer to other designers or from one product to the next. Other researchers have done work to use human input to help make decisions. Michalek and Papalambros demonstrated the use of visual representations of architectural layouts and enable designers to have a level of interactive optimization [17]. This is shown to be useful in architectural layout design because of the subjectivity related to the performance of floor plan designs [18]. A major focus of research in the areas of neural networks and machine learning is to automatically learn to recognize complex patterns and make decisions based on data [19]. Artificial intelligence methods have been used in creative and subjective applications to create art [20] and to recommend music [21] based on prior learning. Recent research using evolutionary computation methods in engineering design acknowledges that there are potential gains from using subjective human evaluation to guide optimization toward better solutions, especially when the problem is less well-defined such as during conceptual design [22]. Similarly, the decisions that designers make in the early phases of the design process involve a complex mix of quantitative *and* qualitative evaluations. Some of these preference capture methods in literature will be used or adapted in the methodology of the present paper, in order to model the subjective preferences of designers and guide the automated search accordingly.

3 Method Development

The main objective of the methodology presented in this paper is to capture designer preferences for design features and incorporate that information into the numerical search that automatically forms and evaluates new designs. The new design methodology has eleven major steps and is described graphically by Fig. 1

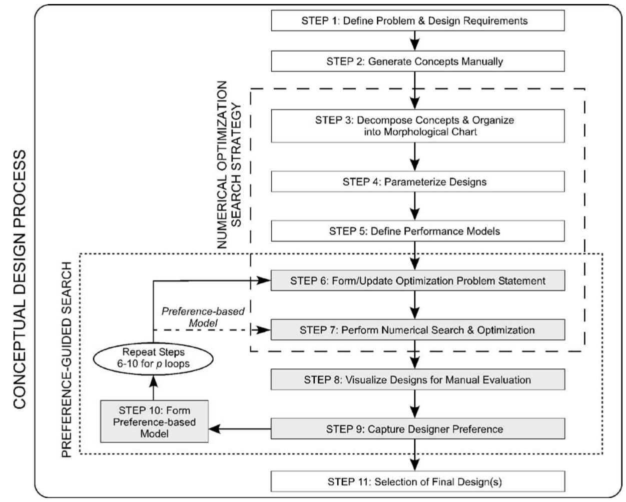

Fig. 1 The conceptual design context within which the preference-guided search methodology fits

within the context of a conceptual design process. Steps 1–5 will be briefly described here but further details can be found in our previous works [6,7]. Steps 6–10, identified in the figure as the *preference-guided search*, contain the preference capture and incorporation methodology, and are the main focus of this paper.

3.1 Step 1: Define Problem and Design Requirements. Defining the design problem and the design requirements, conditions that a concept must satisfy, involves understanding the customer needs and translating them into functional product specifications and design objectives $(\mu_1, \mu_2, \dots, \mu_{n_\mu})$. Design requirements may be quantitative, requiring calculated performance levels, as well as qualitative in nature, using subjective judgment and intuition to evaluate performance. There are numerous effective methods for this step presented in literature [1–4].

3.2 Step 2: Generate Concepts Manually. After defining the design requirements, designers manually generate concepts, using any effective method at his/her disposal, many such methods are discussed in literature [1–4]. Concept descriptions could be as simple as a list of ideas, or as complex as detailed, dimensioned sketches.

3.3 Step 3: Decompose Concepts and Organize Into Morphological Chart. This step involves decomposing each manually generated concept into subfunctions and subfunction solutions, which we will refer to as features, and organizing them into a morphological chart [1,3]. The subfunctions are placed in the first column of each row, and the features are placed in the row of their corresponding subfunction. The purpose of this decomposition method is to categorize features according to their intent so that new concepts can be formed with combinations of interchangeable features from the chart.

3.4 Step 4: Parameterize Designs. The morphological chart is now parameterized, or represented numerically, in a two-dimensional matrix. This matrix is shown generically in Eq. (1) as matrix F_m, which will be referred to as a *morphological matrix*.

$$F_m = \begin{bmatrix} F_{11} & F_{12} & \cdots & F_{1(n_{F_M})} \\ F_{21} & F_{22} & \cdots & F_{2(n_{F_M})} \\ \vdots & \vdots & \ddots & \vdots \\ F_{(n_R)1} & F_{(n_R)2} & \cdots & F_{(n_R)(n_{F_M})} \end{bmatrix} = \begin{bmatrix} 1 & 2 & 3 & \cdots & m_1 \\ 1 & 2 & 3 & \cdots & m_2 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & 2 & 3 & \cdots & m_{n_R} \end{bmatrix} \tag{1}$$

The size of the matrix F_m is $n_R \times n_{F_M}$, where n_R is the number of function rows in the morphological chart and n_{F_M} is the maximum number of features in any of the rows. F_{ij} is treated as a discrete solution, where each i is numbered sequentially from 1 to m_i. Features characterized by continuous numbers are handled differently.

Equation (2) shows matrix F_c, which contains the lower and upper bounds for each continuous variable needed to fully describe designs numerically.

$$F_c = \begin{bmatrix} F_{c_{1_l}} & F_{c_{1_u}} \\ F_{c_{2_l}} & F_{c_{2_u}} \\ \vdots & \vdots \end{bmatrix} \quad (2)$$

The method of parameterization presented herein uses a chromosomelike numerical representation of designs: a column vector of discrete values from each row of F_m and continuous values from within the bounds in each row of F_c, which we will call the *design chromosome*. Generically, the design chromosome is defined here as

$$c = \begin{bmatrix} x_{d_1} & \ldots & x_{d_n} & x_{c_1} & \ldots & x_{c_m} \end{bmatrix}^T \quad (3)$$

where x_d is a set of discrete variables from the morphological matrix F_m and x_c is a set of continuous design variables from F_c. The numerical values contained within the design chromosome, which represent each discrete variable and continuous variable, are defined as the *genes* of a design chromosome. A more thorough description of this parameterization method is provided in our previous works [6,7]. The purpose of parameterization is to enable the use of the manually generated concepts in the numerical search in step 7.

3.5 Step 5: Define Performance Models. The purpose of this step is to define the quantitative performance models, which can each be physics-based models μ^{phys}, or preference-based models μ^{pref}. They will be used to automatically evaluate new designs that are found by the numerical search and optimization in step 7. The output of the performance models is a measure of how well designs meet the design requirements defined in step 1. Each performance model μ is defined as $\mu = f(x_d, x_c)$. The physics-based models are generally simplified approximations of more complex analysis models used in the design embodiment or detailed design phases. These types of calculations are usually the ones performed manually as feasibility calculations or "ball park" performance calculations. However simple, these models can be very powerful when implemented into the numerical search of step 7 because of the vast number of calculations that can be performed automatically and quickly. Initially, the known performance models will likely include only the physics-based models that are applicable to the design problem at hand, and the preference-based model will be interactively formed in the remaining steps.

3.6 Step 6: Form/Update Optimization Problem Statement. The purpose of this step is to form (on the first loop), or update (on subsequent loops), an optimization problem statement that will be used in step 7 to search for the best performing designs. To form a quantitative preference-based model, the physics-based models are initially excluded from the optimization problem statement for a designer-specified number of learning loops. This *learning period* allows the numerical search to explore the design space and gives the designer an opportunity to manually evaluate designs from a wide design space, without the optimizer driving toward optimal designs as defined by physics-based models alone. It is important to note that upon executing step 6 for the first loop, when the designer has yet to indicate preference, the preference function is a constant, which results in a set of randomly generated designs. These designs are manually evaluated in the proceeding steps and the preference function is no longer a constant and is updated based on the preference expressed during the manual evaluation. During the entire learning period, the multiobjective optimization problems reduces to a single objective optimization problem, as shown here

$$\min_x \{-\mu^{\text{pref}}(x_d, x_c)\} \quad (4)$$

subject to

$$g_q(x) \le 0, \quad \forall\, q \in \{1, \ldots, n_g\} \quad (5)$$

$$h_v(x) = 0, \quad \forall\, v \in \{1, \ldots, n_h\} \quad (6)$$

$$x_{jl} \le x_j \le x_{ju}, \quad \forall\, j \in \{1, \ldots, n_x\} \quad (7)$$

where g is a vector of inequality constraints, h is a vector of equality constraints, x is a vector of design variables, and the design variables are bound by their lower (l) and upper (u) limits shown in Eq. (7).

When the learning period ends, a decision that will be discussed later, the optimization problem statement is *updated* to be a multiobjective optimization problem statement, incorporating the physics-based models and the newly formed preference-based models, as shown here

$$\min_x \{-\mu^{\text{phys}}(x_d, x_c), -\mu^{\text{pref}}(x_d, x_c)\} \quad (8)$$

subject to constraints in Eqs. (5)–(7).

The purpose of the optimization problem statement is to direct the numerical search and optimization, in step 7, to find the best performing designs according to those performance models included within the current form of the optimization problem statement. Note that the creation of the preference-based model is explained in detail in step 10 but here it is sufficient to say that once it is formed, it is used in an equivalent manner as the physics-based models.

3.7 Step 7: Perform Numerical Search and Optimization. Guided by the current optimization problem statement, a numerical search and optimization is now performed to find designs that will be presented to designers in step 8. The presence of discrete and continuous variables makes evolutionary search methods, such as genetic algorithms [23], a preferred optimization search strategy for the design methodology presented in this paper. The numerical search begins by creating an initial population of designs, where each design is expressed in the form of concept chromosomes. On the initial loop of the search, the population is created in a random manner. On subsequent loops, a portion, or all, of the designs present at the end of the previous loop can be used in this step. This replicates the inheritance principle of evolutionary algorithms, with the intention of carrying over the best design traits into successive loops in order to continue to improve the performance of the designs generated by the methodology. When created randomly, the discrete genes in the design chromosome are randomly chosen from the integer values in each row of F_m. The genes that correspond to continuous variables are selected randomly from values within their allowable ranges in F_c.

Next, the physics-based models and/or preference-based models, which are included in the current optimization problem statement, are used to evaluate the performance of the current population of designs. These performance values are used to calculate a *maximin* fitness value for each design. The maximin fitness function is used because of the way it directs genetic algorithms to find a diverse set of nondominated (Pareto-optimal) solutions [24]. The maximin fitness value of each design is what determines the probability that a design will be selected to be a parent-design for reproduction.

After fitness values have been calculated for the entire population of designs, the genetic algorithm operations of tournament selection, gene-by-gene crossover reproduction, mutation, and elitism are used to gradually improve the overall performance of the population [23]. This genetic algorithm evolutionary optimization process repeats until a termination condition is met, which can be a specified number of generations or until a convergence criteria has been met, such as a minimum amount of change in objective values from one generation to the next. In the case of the maximin fitness function, the progression of fitness scores from more negative to less negative indicates that the Pareto designs are becoming less clustered. In other words, the Pareto designs are more evenly spread out over the Pareto frontier.

3.8 Step 8: Visualize Designs for Manual Evaluation. After the stopping criteria for the numerical search and optimization in step 7 have been met, a subset of designs from the final generation/population (n_V number of designs) is presented to the human designer for subjective evaluation. The features present in this subset are representative of the population, allowing the designer to consider a diverse set of feature combinations without evaluating dozens of designs. A smart-Pareto filter [25,26] is one way to eliminate designs that are too similar to other designs based on the relative closeness of their objective values. Similar approaches may be used to do the same in variable space, should the designer find this important to do. It may also be advantageous to present the set of designs with the best performance according to the preference-based models, or any of the physics-based models.

In addition to the designs selected through a filtering strategy, a percentage of the designs presented to the designer r_R should be randomly selected. This helps maintain a level of diversity within the set of designs presented, especially during the learning period as the population of designs gradually converges on similar designs that match the designer's preference.

To help a designer quickly comprehend the make-up of the designs being evaluated, the calculated performance values of the physics-based models and preference-based models are presented visually along side graphical representations of designs. The graphical representations can be created through CAD or other parametric software that can quickly generate the visual images. Having the performance data and the images of the designs shown together helps a designer make trade-offs between the quantitative performance and the qualitative aspects of the designs as he/she selects preferred designs.

3.9 Step 9: Capture Designer Preferences. With the visual representations of the designs presented along with the performance levels, a designer now selects the designs that he/she prefers. One purpose for having a human designer manually evaluate designs is to attempt to capture unmodeled objectives. Also, a human can very quickly make mental trade-offs of competing objectives, resulting in subjective decisions. Evaluation methods, such as rating, ranking, or scoring of the designs, could be used to indicate preference. The lowest level of rating is to *select* or *not select* individual designs as preferred, which is the rating/evaluation method used in this paper. In each successive round of manual evaluation, the design chromosomes of the designs, which are selected as preferred, are recorded. The feature frequency count and variable values for each preferred design will be used next, in step 10, to form a quantitative preference-based model.

3.10 Step 10: Form Preference-Based Model. In order to automatically evaluate a designer's preference for designs, which were formed in step 7, a mathematical model is now formed to predict the designer's preference for certain features and parameter values. In order to quickly form a preference model, we use statistical probability as the underlying theory to predict the probability that a design will be preferred by the designer. For both discrete and continuous genes in the design chromosome, the gathered preference data are used to estimate the probability density functions.

When creating an individual preference model for the discrete genes, a *discrete probability density estimate*, also called the probability mass estimate, is created. Figure 2(a) shows an example of a histogram for each value present in the preferred designs from step 9. The histogram is actually a graphical estimate of the real probability density function. The probability estimate $\hat{f}$ for a new gene value x is equal to the proportion of previously recorded genes with the same value [27], as shown here

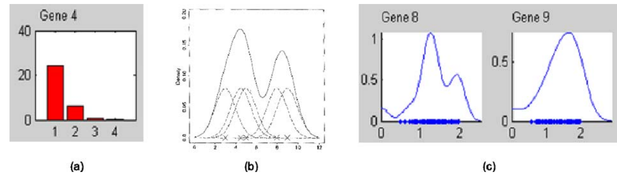

Fig. 2 Examples of (*a*) an estimated probability mass function for a discrete variable, (*b*) a standard kernel density estimate [29], and (*c*) an estimated probability density function for a continuous variable, used to model and predict a designer's preference for the variables. Note that in all cases, the horizontal axis represents the gene values, and the vertical axis represents how preferable a gene value is.

$$\hat{f}(x) = \frac{n_{X_i}}{nh} \tag{9}$$

where n is the number of bins and h is the width of the bins and n_{X_i} is the number of values in the same histogram bin as x.

When creating an individual preference model for the continuous genes, we use density estimation and smoothing techniques [27,28], which empirically form a distribution through the summation of normal distributions around each data point. The gene values present in the preferred designs from step 9 are used in Eq. (10) to form the probability density estimate.

$$\hat{f}(x,h) = \frac{1}{nh}\sum_{i=1}^{n} K\left(\frac{x - X_i}{h}\right) \tag{10}$$

where the kernel K is a distribution satisfying $\int K(x)dx=1$ (i.e., a normal distribution) and h is the *smoothing parameter* [27]. This type of density estimate is illustrated in Fig. 2(b), where the individual kernels (the small distribution curves placed over each data point) are summed up to form the density estimate $\hat{f}$ [29]. As with the histogram, the smoothness of the curve is determined by the bandwidth, or smoothing parameter h, of the individual kernels. The *normal optimal smoothing* method is one of the most common and most effective methods to choose a smoothing parameter [28]. Assuming the kernel K is a normal density, the smoothing parameter h is calculated as

$$h = \left(\frac{4}{3n}\right)^{1/5}\sigma \tag{11}$$

where σ is the standard deviation of the distribution [28].

Figure 2(c) shows an example of a probability density estimate for a continuous variable and the points used to create it. This empirical approach to create the preference function allows the model to be updated each time evaluations are completed by the designer, thus continually improving the accuracy of the model.

The individual preference models for each gene in the design chromosome can be combined into a single preference model used to predict preference for an entire design, as defined here

$$\mu_m^{\text{pref}} = (\hat{f}_{x_{d_1}}(x_{d_1}) \times \cdots \times \hat{f}_{x_{d_i}}(x_{d_i}) \times \hat{f}_{x_{c_1}}(x_{c_1}) \times \cdots \times \hat{f}_{x_{c_j}}(x_{c_j})) \tag{12}$$

where $\hat{f}_{x_{d_i}}$ is the probability density estimate for the ith discrete variable/gene in a design chromosome and $\hat{f}_{x_{c_j}}$ is the probability density estimate for the jth continuous variable/gene in a design chromosome. This combined preference model μ_m^{pref}, after being sufficiently developed, can be used as a preference-based objective in the optimization problem statement, when it is updated in step 6, to guide the numerical search toward designs with a higher probability of being preferred by the designer.

3.11 Step 11: Selection of Final Design(s). Each loop of the preference-guided search in steps 6–10 gathers more data with which it can update the preference-based model, gradually improving the ability of the numerical search to find designs that will be preferred by the designer while also including the physics-based evaluations. It is generally accepted in the field of interactive evolutionary computation (IEC) research that when subjective human preference is involved, there is not a global optimum represented by a single design [30]. For this reason, the new computationally assisted design methodology attempts to thoroughly search for a set of designs in a global optimum area. When a designer is satisfied with the final set of designs, any number of designs from that set can be used as a starting point for the next phase of the product development process or as inspiration for further conceptual design efforts.

During the learning period, the preference-based model will be updated and improved with each loop of steps 6–10, and the search will gradually converge on a set of similar designs that match the preference of the designer. The decision to end the learning period can be made by the designer when he/she feels that the individual gene preference models have been accurately captured, or a stopping criteria has been met, such as when all of the designs selected for visualization (see step 8), which are not randomly selected $n_F = n_V(1-r_R)$, have preference-based performance scores above a designer-specified amount, such as 60%. In loops after the learning period, stopping criteria could be determined in a similar fashion as an alternative to stopping at the discretion of the designer.

4 Vehicle Architecture Example

In this section, a vehicle architecture example is provided to illustrate the 11-step method presented in Sec. 3. This section is divided into two parts: (a) moving through the 11 steps for the vehicle example and (b) a test of the method to demonstrate convergence on more preferred designs.

4.1 Example: Vehicle Architecture Design. Periodically, automobile manufacturers will produce a new type of vehicle that is considered innovative, by changing the form and/or function to a new look or use. Several modern examples are vehicle types such as the minivan, the station wagon, the sport utility vehicle, and the crossover. Realistically, these vehicles are simply different combinations of existing features and parameters of other vehicles. Albeit designs based on recombination, there was a market for those types of new vehicles, which may have been based on functionality, aesthetic appeal, or some combination. While fairly simple outwardly, automobiles are very complex systems on the inside, using multiple, high-tech, integrated mechanical and electrical systems that functionally perform as specified. In this section, the computationally assisted design methodology is applied to a vehicle architecture design example, and used to demonstrate the ability of the methodology to (1) capture designer preference for features and parameters of vehicle design and (2) form and incorporate quantitative preference-based models with physics-based performance models into the search for, and evaluation of, significantly more designs and novel designs than could be done by manual methods.

Table 1 Design objectives for vehicle design example

	Objective	Units	Direction	Range
1	Price, μ_1	\$	Minimize	$0<\mu_1$
2	Weight, μ_2	lbs	Minimize	$0<\mu_2$
3	Seating, μ_3	n/a	Maximize	$0<\mu_3$
4	Towing, μ_4	lbs	Maximize	$0<\mu_4$
5	Cargo space, μ_5	ft^3	Maximize	$0<\mu_5$
6	Preference, μ_6	n/a	Maximize	$0<\mu_6<1$

4.1.1 Example: Steps 1, 2, and 3, Define Requirements, Generate, and Decompose Concepts. A list of the vehicle architecture design objectives and their boundary conditions is shown in Table 1. Figure 3 shows examples of the human-generated concept sketches for current vehicle types such as the compact car, midsize sedan, pick-up truck, passenger van, and so forth. The functions of these vehicle concepts, along with other common vehicle features and components, have been decomposed into the morphological chart, shown in Table 2. Note that FWD, RWD, AWD, and 4WD denote front, rear, all, and four-wheel drive, respectively. Additional design variables are shown in Table 3. The design features and other design variables can be seen in Fig. 4, which contains a schematic of the vehicle architecture design.

4.1.2 Example: Step 4, Parameterize Designs. The numerical form of the morphological chart is shown in Eq. (13) as the morphological matrix F_m.

$$F_m = \begin{bmatrix} F_{11} & F_{12} & & & & \\ F_{21} & F_{22} & F_{23} & F_{24} & F_{25} & \\ F_{31} & F_{32} & F_{33} & F_{34} & F_{35} & F_{36} \\ F_{41} & F_{42} & F_{43} & F_{44} & & \\ F_{51} & F_{52} & F_{53} & & & \end{bmatrix} = \begin{bmatrix} 1 & 2 & & & & \\ 1 & 2 & 3 & 4 & 5 & \\ 1 & 2 & 3 & 4 & 5 & 6 \\ 1 & 2 & 3 & 4 & & \\ 1 & 2 & 3 & & & \end{bmatrix} \tag{13}$$

The complete design chromosome is defined as

$$c_{\text{vehicle}} = [F_1 \ \ F_2 \ \ F_3 \ \ F_4 \ \ F_5 \ \ w_B \ \ n_L \ \ t_D \ \ s_{H2} \ \ n_r \ \ t_r \ \ g_C]^T \tag{14}$$

4.1.3 Example: Step 5, Define Performance Models. The five physics-based performance models, price (μ_1), weight (μ_2), seating (μ_3), towing capacity (μ_4), and cargo space (μ_5), along with

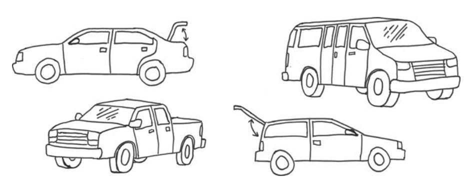

Fig. 3 Human-generated concept sketches of vehicles

Table 2 Morphological chart for vehicle architecture design example

Feature	Possible solutions					
Doors, F_1	Two doors	Four doors				
Chassis, F_2	Compact	Midsize	Full-size	Heavy duty	Super duty	
Engine, F_3	Four-cylinder	V6	V8	V8 diesel	Electric	Hybrid
Drive type, F_4	FWD	RWD	AWD	4WD		
Cargo style, F_5	Rear hatch	Truck bed	Trunk			

Table 3 Design variables for vehicle architecture design example

Variable	Possible values
Wheel base, w_B	72–175 in. (4.45 mm)
Nose length, n_L	25–50 in. (0.64–1.27 mm)
Tire diameter, t_D	20–36 in. (0.51–0.91 mm)
Cab height, s_{H2}	36–120 in. (3.05 mm)
No. of seat rows, n_r	1, 2, 3, 4
Tire aspect ratio, t_r	0.20–0.85
Ground clearance, g_C	6–20 in. (0.15–0.51 mm)

the unmodeled preference-based performance model for aesthetics (μ_6), are defined in this section. The first is price and is defined as

$$\mu_1 = m_B C_m + C_t + C_{F2} + C_{F3} \tag{15}$$

where m_B represents the mass of the body, C_m represents the cost of the body material, C_t represents the cost of the tires, C_{F2} represents the cost of the chassis type, and C_{F3} represents the cost of the engine type. Specifically

$$m_B = \sum_{i=2}^{4} 0.283\alpha_i (L_{si} H_{si} W_{si}) \tag{16}$$

where L_{si}, H_{si}, and W_{si} are the length, height, and width of the ith segment of the vehicle and

$$\alpha_i = \begin{cases} 0.00067, & i = 2,3 \\ 0.00333, & i = 4 \end{cases} \tag{17}$$

and

$$C_m = 3.15 \ \$/\text{lbs} \tag{18}$$

$$C_t = 20 t_D \tag{19}$$

$$C_{F2} = 0.283 L_f H_f W_f C_m (0.02 + F_4/10) \tag{20}$$

$$C_{F3} = \begin{cases} 800, & \text{four-cylinder} \\ 1000, & \text{V6} \\ 1200, & \text{V8} \\ 1400, & \text{V8 diesel} \\ 1600, & \text{electric} \\ 1400, & \text{hybrid} \end{cases} \tag{21}$$

where t_D is the tire diameter and L_f, H_f, and W_f represent the length, height, and width of the frame.

The second objective is weight and is defined as

$$\mu_2 = m_B + m_t + m_{F2} + m_{F3} \tag{22}$$

where

$$m_t = 8 t_D \tag{23}$$

$$m_{F2} = 150 F_2 + 550 \tag{24}$$

$$m_{F3} = \begin{cases} 600, & \text{four-cylinder} \\ 800, & \text{V6} \\ 1000, & \text{V8} \\ 1200, & \text{V8 diesel} \\ 1000, & \text{electric} \\ 1100, & \text{hybrid} \end{cases} \tag{25}$$

The third objective is seating capacity and is simply defined as

$$\mu_3 = 3 n_r \tag{26}$$

The fourth objective is towing capacity, which is defined as

$$\mu_4 = \begin{cases} 0, & F_3 = 5,6 \\ 1000(F_2 - 1)\sqrt{F_2(F_3 - 1)}, & \text{otherwise} \end{cases} \tag{27}$$

The fifth objective is cargo space and is defined as

$$\mu_5 = c_r V_c \tag{28}$$

where

$$V_c = L_{s4} H_{s4} W_{s4} \tag{29}$$

and

$$c_r = \begin{cases} 0.50, & \text{hatch} \\ 0.70, & \text{truck bed} \\ 0.25, & \text{trunk} \end{cases} \tag{30}$$

and the sixth objective is aesthetics, which is defined in this paper as

$$\mu_6 = f_{\text{aesthetic}}(x_d, x_c) \tag{31}$$

where x_d is a set of discrete variables/genes and x_c is a set of continuous variables/genes that describe a design.

4.1.4 Example: Step 6, Form/Update Optimization Problem Statement. The single objective optimization problem statement used during the learning period of the preference-guided search uses only the preference-based model μ_6 and is equivalent to Eq. (4). To be clear, the smoothing parameter h is calculated for the example according to Eq. (11). The actual calculated value that is used for each loop of the process depends on the parameter n (see Eq. (11)), which is equal to the total number of designs selected by the designer throughout all loops combined. As the value for the parameter n increases, the smoothing parameter provides a more accurate description of the designer's preference. The multiobjective optimization problem statement used after the learning period is over contains the physics-based models and the newly formed preference-based model, resulting in the following form of Eq. (8), as shown here

$$\min_x \{\mu_1(x_d,x_c), \mu_2(x_d,x_c), -\mu_3(x_d,x_c), -\mu_4(x_d,x_c), -\mu_5(x_d,x_c), -\mu_6(x_d,x_c)\} \tag{32}$$

subject to

$$0 < \mu_1 \le 100{,}000 \ \$ \tag{33}$$

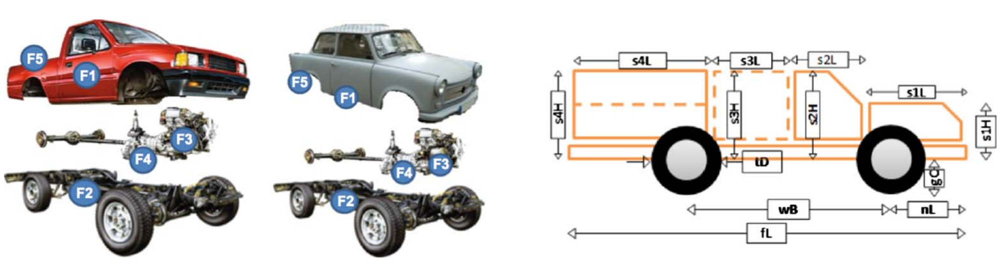

Fig. 4 A schematic of the vehicle architecture design example

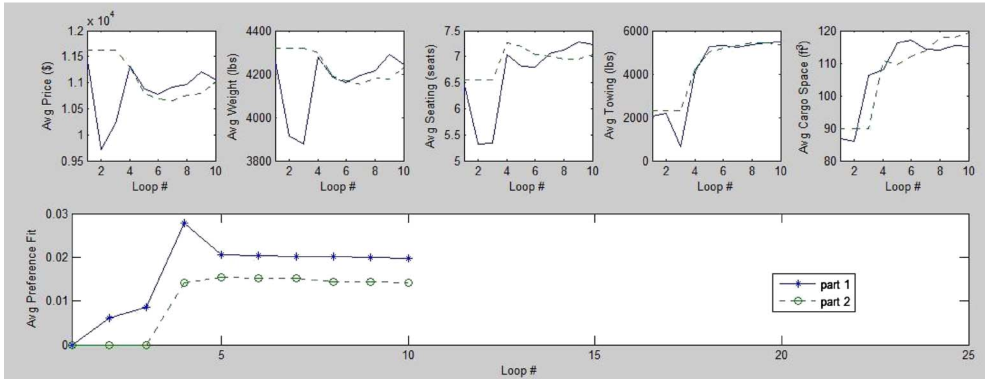

Fig. 5 The progression of the design objectives through the preference-guided search, changing from single objective optimization to multiobjective optimization after loop 3

$$0 < \mu_2 \leq 50{,}000 \text{ lbs} \tag{34}$$

$$2 \leq \mu_3 \leq 15 \text{ seats} \tag{35}$$

$$0 < \mu_4 \leq 26{,}000 \text{ lbs} \tag{36}$$

$$20 \leq \mu_5 \leq 200 \text{ ft}^3 \tag{37}$$

$$0 < \mu_6 \leq 1 \tag{38}$$

where x_d is a set of discrete variables/genes and x_c is a set of continuous variables/genes that describe a design.

4.1.5 Example: Step 7, Perform Numerical Search and Optimization. The genetic algorithm used as the numerical search method used a generation size of $N=540$, a crossover probability of $p_{\text{crossover}}=0.2$, a mutation probability of $p_{\text{mutation}}=0.0001$, a tournament ratio of $r_{\text{tournament}}=0.2$, and number of generations $G=20$. These conditions for the numerical search result in the automatic evaluations of 10,800 designs/combinations of features and parameters per execution of the numerical search in each loop of the design methodology.

Figure 5 shows the progression and convergence of the objectives in the numerical search over ten loops. Here, there are two populations that are being evaluated, labeled part 1 and part 2. Part 1 incorporates the newly formed preference-based model, and part 2 does not. Both populations are observed so that the affects of including the preference-based model can be seen. The first three loops were the learning period, when only the preference-based objective was used. After the preference-based model was developed, the physics-based objectives were included in the numerical search for designs, changing to a multiobjective search and causing trade-offs to be made between the competing objectives. After ten loops, all the designs were nondominated, Pareto-optimal designs, and the objectives had converged, having found a set of designs that best met the objectives.

4.1.6 Example: Step 8, Visualize Designs for Manual Evaluation. Figure 6 shows visual representations of a set of vehicle designs that were automatically formed, through random selection methods during the learning period of the design methodology when no optimization has taken place. The visual representations of the vehicles also present each design's physics-based and preference-based performance levels, and several critical design gene/feature descriptions.

4.1.7 Example: Steps 9 and 10, Capture Designer Preferences and Form Preference-Based Model. After a designer subjectively evaluates the displayed designs, the features and parameters present in the preferred designs are captured and used to form the preference-based models. Examples of the preference models formed are shown in Fig. 7, where those shown as histograms are for discrete genes and those with probability density functions are for continuous genes. Figure 8 shows a set of designs presented at the end of the learning period. Having been optimized according to the newly formed preference model, these designs are generally similar in appearance but have different trade-offs of physics-based performance.

4.1.8 Example: Step 11, Selection of Final Design(s). Figure 9 shows a representative set of designs from the final loop, after the optimization has included the new preference-based model and the physics-based models.

4.2 Example Validation. To validate that the new method indeed identifies more-and-more designs that are designer-preferred, an automated test was performed. The test ran through the steps of the methodology, automatically selecting designs in step 9 according to predetermined preference criteria in order to simulate the choices that a human designer would make. In this test, there are two populations that are being evaluated, labeled part 1 and part 2. Part 1 incorporates the newly formed preference-based model, and part 2 does not. Both populations are observed so that the affects of including the preference-based model can be seen. The upper plot in Fig. 10 shows an improvement in the average preference-based performance through the ten loops of the learning period. Notice that a more negative number indicates that a population is "more preferred." Also notice that when the physics-based models are added (loop 11), the preference is worsened as the optimizer seeks to also optimize the physics-based objectives. The bar chart shows that during the learning period, there are progressively more preferred designs

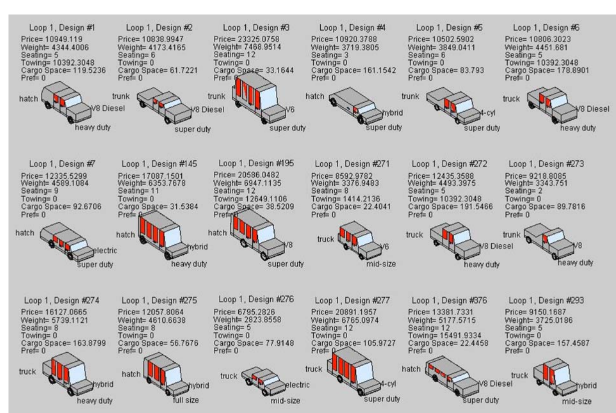

Fig. 6 Set of nonoptimized vehicle designs automatically formed and presented to human designers for subjective evaluation

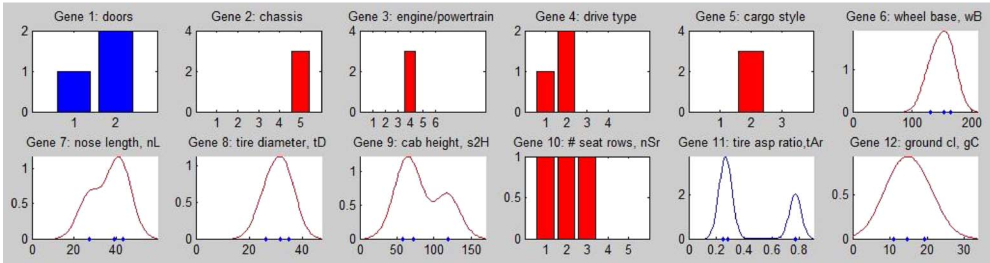

Fig. 7 The preference-based models for each design gene, formed from subjective evaluation of a designer. Compare with Fig. 2.

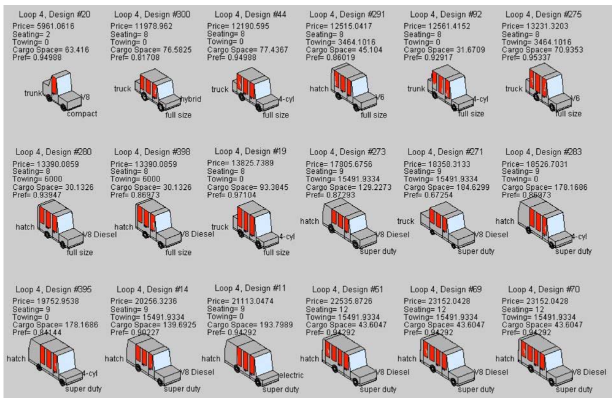

Fig. 8 A set of vehicle designs that has converged using the preference-based models and physics-based models of performance

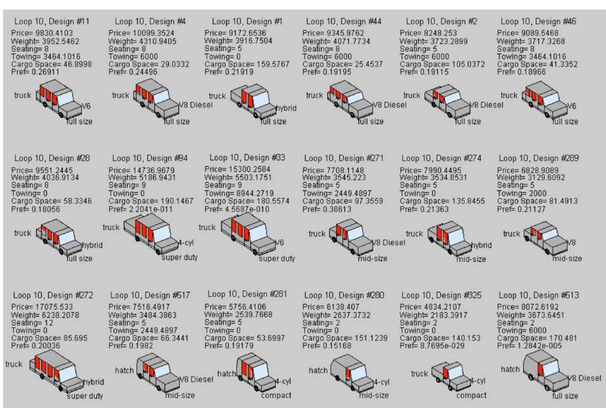

Fig. 9 New, optimized vehicle designs automatically formed, consisting of new combinations of features and parameter values

being identified in part 1 population, as compared with part 2 population. Notice that over the ten loops of the learning period, the quantity of preferred designs from part 1 steadily increases from 2 to 9 while the quantity of preferred designs from part 2 varies inconsistently between 2 and 6, as indicated by the area of the lightly shaded region of the bar chart.

This example, the results, demonstrated the use of all the steps of the design methodology to search through vast number of possible combinations of design features and parameters (10,800 per loop), and converge on a set of preferred designs, as shown in Fig. 9. The vehicle features present in these designs are recognizable from the initial concept sketches (see Fig. 3), however, many of the combinations of features are new and were not previously considered by the designer.

5 Concluding Remarks and Limitations

This paper has focused on improving the set of designs that is considered during conceptual design, by rapidly exploring design possibilities and by incorporating human-based subjective evaluation into a computational search. It was shown that the new design methodology uses an interactive, statistics-based, preference capture method to form a quantitative model of the subjective design decisions a designer made during the manual evaluation of designs. This preference-based model, along with the physics-based models, was used with multiobjective optimization to guide the numerical search for designs that are nondominated, the match the *preference* of the designer, and that represent *new combinations* of features and parameter values that may not have been found through manual methods. Importantly, the methodology was able to rapidly evaluate tens of thousands of designs per minute.

Future work to improve the effectiveness of the subjective evaluation done by the designer could include improvements in the visualization of design composition and performance. The preference capture methods in this paper also do not account for combinatorial effects or unequal weighting of preferred features

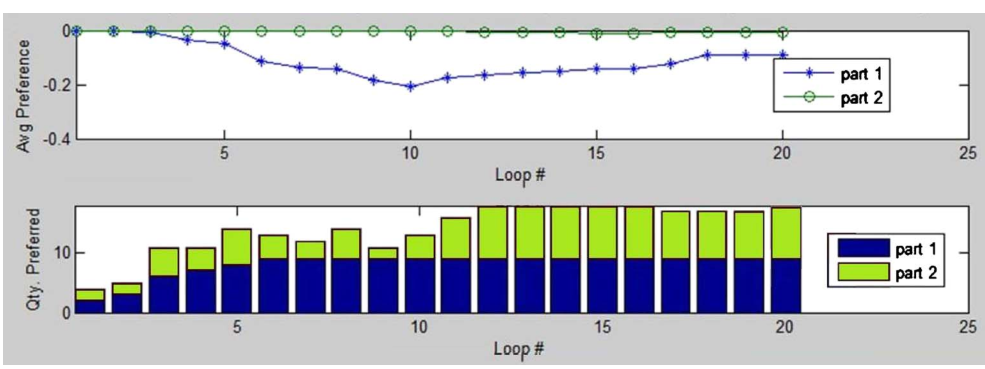

Fig. 10 Test data showing the preference model improvements during the learning period (top), and a higher quantity of preferred designs being found (bottom)

and performance levels. There also exists the challenge of how to model performance of designs when unfamiliar combinations of features occur that are not covered by the existing physics-based models. These new combinations can either be treated as infeasible designs, or as an opportunity to develop new performance models, potentially leading to the discovery of innovative products.

It is likely that the application of this methodology will be most successful for design teams that repeatedly redesign the same type of products because they will have access to, or the expertise to create, well developed physics-based models. It is also well suited for the design of modular products. The methodology could also assist these same teams in more fully exploring the design space and finding designs that accomplish their existing design requirements with new combinations of features that had not previously been considered. Considering this, the methodology could be a useful conceptual design tool in industries such as consumer products, automotive, consumer electronics, recreational products, or any product that has the combination of qualitative design requirements and quantitative, performance requirements. We note that it would be more challenging to use this methodology when the design requirements are abstract and the goal is to reinvent the way tasks are approached.

Acknowledgment

We would like to recognize the National Science Foundation under Grant No. CMMI-CAREER-0954580 for funding this paper. We would also like to acknowledge Jacob Montierth of Ford Motor Company who provided input for the models used in the vehicle architecture example.

References

[1] Pahl, G., and Beitz, W., 1996, *Engineering Design: A Systematic Approach*, Springer, New York.
[2] Ullman, D., 2002, *The Mechanical Design Process*, McGraw-Hill, New York.
[3] Ulrich, K., and Eppinger, S., 1995, *Product Design and Development*, McGraw-Hill, New York.
[4] Pugh, S., 1990, *Total Design*, Addison-Wesley, London.
[5] Otto, K., 1995, "Measurement Methods for Product Evaluation," Res. Eng. Des., **7**(2), pp. 86–101.
[6] Barnum, G., and Mattson, C., 2009, "A Numerical Optimization Search Strategy for Exploring Morphological Charts," 50th AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference, American Institute of Aeronautics and Astronautics.
[7] Barnum, G., 2010, "A Computationally-Assisted Methodology for Rapid Exploration of Design Possibilities in Conceptual Design," MS thesis, Brigham Young University.
[8] Huang, G. Q., and Mak, K. L., 1999, "Web-Based Morphological Charts for Concept Design in Collaborative Product Development," J. Intell. Manuf., **10**(3–4), pp. 267–278.
[9] Stone, R., Wood, K., and Crawford, R., 1999, "Product Architecture Development With Quantitative Functional Models," Proceedings of the Design Engineering Technical Conferences.
[10] Bryant, C., Stone, R., McAdams, D., Kurtoglu, T., and Campbell, M., 2005, "SA Computational Technique for Concept Generation," Proceedings of the DETC2005, Paper No. DETC05/DTM-85323.
[11] Kopena, J., and Regli, W., 2003, "Functional Modeling of Engineering Designs for the Semantic Web," Bulletin of the Technical Committee on Data Engineering, **26**(4), pp. 55–62.
[12] Bryant, C., Stone, R., McAdams, D., Kurtoglu, T., and Campbell, M., 2006, "A Validation Study of an Automated Concept Generator Design Tool," Proceedings of the DETC/DTM2006, Paper No. DETC2006-99489.
[13] Bryant, C., Bohm, M., Stone, R., and McAdams, D., 2007, "An Interactive Morphological Matrix Computational Design Tool: A Hybrid of Two Methods," Proceedings of the DETC/DTM2007, Paper No. DETC2007-35583.
[14] Hutcheson, R., Jordan, R., Jr., Stone, R., Terpenny, J., and Chang, X., 2006, "Application of a Genetic Algorithm to Concept Variant Selection," Proceedings of the DETC '06 (Design Engineering Technical Conferences), pp. 2006–99390.
[15] Landon, M. D., and Balling, R. J., 1994, "Optimal Packaging of Complex Parametric Solids According to Mass Property Criteria," ASME J. Mech. Des., **116**, pp. 375–381.
[16] Yin, S., and Cagan, J., 2000, "An Extended Pattern Search Algorithm for Three-Dimensional Component Layout," ASME J. Mech. Des., **122**, pp. 102–108.
[17] Michalek, J., and Papalambros, P., 2002, "Interactive Design Optimization of Architectural Layouts," Eng. Optimiz., **34**(5), pp. 485–501.
[18] Michalek, J., Choudhary, R., and Papalambros, P., 2002, "Architectural Layout Design Optimization," Eng. Optimiz., **34**(5), pp. 461–484.
[19] Mitchell, T., 1997, *Machine Learning*, WCB/McGraw-Hill, New York, p. 368.
[20] Norton, D., 2009, "Learning About Creativity From an Artificial Artist," Proceedings of the Seventh ACM Conference on Creativity and Cognition, ACM, New York, NY, pp. 339–340.
[21] Dickerson, K. B., and Ventura, D., 2009, "Music Recommendation and Query-by-Content Using Self-Organizing Maps," IEEE-INNS-ENNS International Joint Conference on Neural Networks, pp. 705–710.
[22] Parmee, I., 2001, *Evolutionary and Adaptive Computing in Engineering Design*, Springer-Verlag, London.
[23] Deb, K., 2001, *Multi-Objective Optimization Using Evolutionary Algorithms*, Wiley, New York.
[24] Balling, R., 2003, "The Maximin Fitness Function; Multiobjective City and Regional Planning," Proceedings of the EMO, Vol. 3, pp. 1–15.
[25] Mattson, C., Mullur, A., and Messac, A., 2004, "Smart Pareto Filter: Obtaining a Minimal Representation of Multiobjective Design Space," Eng. Optimiz., **36**(6), pp. 721–740.
[26] Mattson, C. A., and Messac, A., 2003, "Concept Selection Using s-Pareto Frontiers," AIAA J., **41**(6), pp. 1190–1198.
[27] Silverman, B., 1998, *Density Estimation for Statistics and Data Analysis*, Chapman and Hall/CRC, London.
[28] Bowman, A., and Azzalini, A., 1997, *Applied Smoothing Techniques for Data Analysis: The Kernel Approach With S-Plus Illustrations*, Oxford University Press, Clarendon.
[29] Wand, M., and Jones, M., 1995, *Kernel Smoothing*, Chapman and Hall/CRC, New York.
[30] Takagi, H., 2001, "Interactive Evolutionary Computation: Fusion of the Capabilities of EC Optimization and Human Evaluation," Proc. IEEE, **89**(9), pp. 1275–1296.