Accepted to the ASME Journal of Mechanical Designs, July 2000

A PERFORMANCE-BASED REPRESENTATION FOR ENGINEERING DESIGN

Liang Zhu and David Kazmer
Department of Mech. & Ind. Engineering
University of Massachusetts Amherst

ABSTRACT

A design representation is developed to model multi-attribute systems utilizing multi-dimensional clipping and transformation algorithms. Given a linear system characterization, three types of supporting information is generated for the decision maker: 1) a function matrix that describes the performance attributes dependent upon the decision variables; 2) a decision space that corresponds to the feasible decision set that meets performance requirements, and; 3) a performance space that represents the feasible performance region and the Pareto Optimal set. The analytical method developed for solving these feasible spaces is described for the linear model. A case study is presented to demonstrate how to utilize the representation to locate a feasible solution and proceed to the desired trade-off of multiple attributes. Moreover, the potential incorporations of the representation with other influential design methodologies are discussed.

KEYWORDS

Engineering Design Representation, Multiple Attributes, Constraint Based Design, Quality Function Deployment, Systems Design

1 INTRODUCTION

Engineering design is an unstructured but logic-based process where successive iterations of synthesis and analysis eventually converge to an accepted solution (Paz-Soldan and Rinderle 1989). This iterative process is necessary since it is normally infeasible to deduce the final design from the initial set of specifications. There are many difficulties in the design process that prevent direct convergence to a final design.

First, there are many potential concept strategies that may result in an adequate solution. The development and validation of each design concept requires the configuration of the components within the engineered system and identification of the design parameters. If the approach appears promising, quantitative relationships may be established between the design parameters and performance attributes. These relationships can be complex and numerous such that the designer may not be able to simultaneously navigate and resolve an adequate solution. At this stage, the designer may become acutely aware of difficulties with the design: inadequate performance, conflicting specifications, and infeasibility. The designer may accept the current design as final, modify the design parameters to negotiate between the different performance attributes, consider the relaxation or constriction of the design specifications, or start again with another concept design. This complex decision-making process prevents direct development of the final design in all but incremental redesigns.

This paper describes a design representation that resolves the full set of relationships between multiple design parameters and multiple performance attributes. The described approach solves a generalized constraint based design problem to provide a set of hyperspaces from which multiple half-spaces are utilized to determine the design feasibility, performance trade-offs, and parametric solutions. While the performance-based representation is embodied in a form similar to Quality Function Deployment (QFD), its rational basis is compatible with other influential design methodologies. As such, further discussion of related approaches is warranted with comparisons to the developed performance-based representation.

1.1 Constraint Based Reasoning

Design can be viewed as a process of constraint identification and satisfaction. Unfortunately, the number, non-linearity, and coupling between the constraints hinder the process of searching for a feasible design. Constraint based reasoning seeks to support design decisions by differentiating constraints that critically impact the design from those that are irrelevant. Significant research has resulted in facilitating reasoning about the design's potential performance and physical embodiment. (Rinderle and Krishnan 1990) proposes an interval analysis augmented with monotonicity and dominance principles to identify dominant, active and redundant constraints that enable the designer to clearly perceive critical design considerations. (Thornton 1996) utilized symbolic methods and adaptive searches to identify design feasibility, subject to a set of highly coupled and non-linear constraints. (Kusiak, Wang et al. 1996) utilized similar constraint based approaches to assist in negotiation of constraints from multiple decision maker perspectives.

The performance-based representation is a constraint based approach, which utilizes clipping and convex hull algorithms to form the feasible design hyperspace. The described research extends prior work in constraint based reasoning by explicitly providing the designer with full access to the feasible performance regions and allowable designs. As such, the designer can quickly identify the dominant constraints, evaluate the significance of newly asserted constraints, and permit iterative design evolution and constraint modification to attain an acceptable design.

1.2 Functional Representation

Functional representations seek to formally define the system performance requirements, and then apply knowledge and logic to reason about the system decomposition and physical embodiment. One approach to functional representation is to decompose the system into sub-functions to yield a functional graph that approximates subassembly boundaries (Umeda and Tomiyama 1997). An alternative approach is to utilize syntactic languages to describe a design artifact. Causal Functional Representational Language (CFRL) defined the function with a triplet $\{D_F, C_F, G_F\}$, in which D_F denotes the part whose function is F; C_F is the context where the part is to function; and G_F indicates the description of the functional goal to be accomplished (Iwasaki and al. 1993; Sturges, O'Shaughnessy et al. 1996).

In essence, functional representations decompose the customer requirements and map the functionality to the design features. These methods are targeted at conceptual design, yet lack analysis of design parameter correlation and performance dependency. The predominantly qualitative analysis in these approaches also prevents the implemented system from leveraging the available computation tools at the highest efficiency. However, continued research and pending standardization within CAD systems (Szykman, Racz et al. 1999) indicate the potential for facilitating concept development and quantitative performance modeling (Stone and Wood 1999). The performance-based representation is intended to provide a quantitative analysis for concepts generated through functional representations.

1.3 Quality Function Deployment

Quality function deployment (QFD) seeks to convert customer demands into quality characteristics and systematically develop a plan for the deployment of the finished product (Akao 1990). The most used structure in QFD is the House of Quality (HOQ), whose open structure dramatically integrates different design components together. Unfortunately, the QFD method is not sufficiently rigorous or quantitative. Though crisp specifications may be used, the performance evaluations are typically only estimated. The relations between design parameters and performance attributes are qualitatively defined, and final design evaluation is performed by a sum of weighted scores. While useful in initial deployment, QFD is rarely utilized as a configuration and parameter optimization method.

The performance-based representation was conceived as a rational, quantitative, and dynamic alternative to QFD. Rather than utilize qualitative relations and performance estimates, the performance-based representation requires analytic functions. This greater information content allows the direct evaluation of performance attributes from the selected design parameters. The subsequent design evaluation – regarding feasibility and trade-offs – is automated via rational mathematical algorithms. Moreover, the representation is dynamic, permitting the adaptation of specifications, the evolution of functional relationships, and the addition/deletion of design parameters and constraints. As such, the performance-based representation is not derived from QFD, but allows straightforward transition from QFD to rational engineering practice.

1.4 Design Evaluation Approaches

Significant research has been performed in design evaluation methods including decision based design, robust design, and axiomatic design approaches. Decision based design views the design process as a set of decisions, where a decision is defined as an irrevocable allocation of resources and a choice taken from a set of options (Hazelrigg 1998). Research in DBD seeks to maximize the value of the design by system characterization and parameter selection according to rational, economic and mathematical principles. In comparison, robust design seeks to maximize the product quality by reducing the sensitivity of the performance attributes to uncontrolled variation (Taguchi and Konishi 1992). Finally, axiomatic design seeks to maximize the likelihood of developing an acceptable product by maintaining the independence of performance attributes and minimizing the system information content (Suh 1990).

Each of these approaches provides a methodology for product development as well as metrics for evaluation of the design's aggregate performance. The performance-based representation does not promote any of these methodologies. However, the representation is amenable to all these methodologies. The benefits of axiomatic design strategies become readily apparent to the designer using the representation. Moreover, DBD and robust design evaluations can be incorporated into the design representation to assist in product evaluation. These concepts are currently being explored and will be presented in subsequent papers.

2 REPRESENTATION

For most engineering problems, the objective of the design process is to find a feasible solution which satisfies product specifications. Based on the complexity of the problem, the solution is also expected to exhibit a reasonable trade-off between performance, unit cost and development time. Unfortunately, multiple performance attributes are closely coupled in most applications, obstructing the explicit expression of the design objective function. In another words, no single-step optimization can universally assure the satisfactions of all performance attributes. As such, a design representation is developed in this section to aid the designer to manage the parameters and constraints throughout the design process.

The three main components of the performance-based representation are shown in Figure 1. While the function matrix represents the system relations between design parameters and performance attributes, the decision space and performance space further explicate the mutual relations between design parameters and performance attributes, respectively. Together this set of information provides a full constraint based model of the design feasibility.

2.1 Function Matrix

Each well-defined design objective is one performance attribute. The performance attribute plus the expected satisfaction limits constitutes a specification. Denoting the *i*_th performance attribute as y_i, a typical specification can be expressed as $LSL_i \leq y_i \leq USL_i$. Without loss of generality, a one-sided specification can be formed by substituting -∞ or +∞ to the unspecified limits.

Suppose $y_i = f_i(\boldsymbol{X})$, where $\boldsymbol{X}$ is the design vector, $\boldsymbol{X} = \{x_1, x_2, ..., x_j, ..., x_n\}$ and $LCL_j \leq x_j \leq UCL_j$. Given other design parameters of x_j constant, the sensitivity $y_i = f_i(x_1^c, x_2^c, ..., x_j, ..., x_n^c)$ can be plotted. One example is shown in Figure 2. In this paper, a linearization of the function is applied to acquire the analytical feasible decision space and performance space. Figure 2 shows that an error δ exists when the performance attribute is substituted by its dashed linear approximation. The estimation error δ can be assessed as:

$$\delta = f_i(X) - [f_i(X^C) + \left.\frac{\partial f_i}{\partial x_j}\right|_{X=X^C} \cdot (x_j - x_j^C)] \qquad (1)$$

Depending on the complexity of problem and the controlled range of the design parameter, the estimation error varies significantly. Assuming the relative estimation error is ignorable (less than 10% in Figure 2), the graphical matrix of the design sensitivity essentially becomes the Jacobian determinant J,

$$J = \left.\begin{Bmatrix} \frac{\partial f_1}{\partial x_1} & \frac{\partial f_1}{\partial x_2} & \cdots & \frac{\partial f_1}{\partial x_n} \\ \frac{\partial f_2}{\partial x_1} & \frac{\partial f_2}{\partial x_2} & \cdots & \frac{\partial f_2}{\partial x_n} \\ \cdots & \cdots & \cdots & \cdots \\ \frac{\partial f_m}{\partial x_1} & \frac{\partial f_m}{\partial x_2} & \cdots & \frac{\partial f_m}{\partial x_n} \end{Bmatrix}\right|_{X=X^C} \qquad (2)$$

where X^C is the current design vector. This Jacobian may be derived from analytic functions, numerical simulation, or response surface methods utilizing functional evaluation.

2.2 Decision space

The decision space illustrates the feasible regions for all design parameters within which the specifications are satisfied. Suppose that each specification limit (*LSL* is used without loss of generality, similar for *USL*) is a hyper-face *hf*,

$$hf = \{X \in R^n \mid f(X) = LSL\}, \qquad (3)$$

where R^n is an *N*-dimensional Euclidean space. Therefore, a specification is the half-space *hs* defined by the hyper-face *hf*, such that

$$hs = \{X \in R^n \mid f(X) \geq LSL\}. \qquad (4)$$

As such, the decision space is composed by the intersection of all specification half-spaces.

In order to show the hyper-dimensional decision space, the described representation utilizes a set of view panes, each of which corresponds to one mutual relation of design parameters. Therefore, an *N*-dimensional space can be substituted by *(N-1)(N/2)* 2-dimensional panes. Each 2-dimensional space can be further simplified by fixing the other parameters at the constant level X^C. As such, the specification can be plotted in the design space, as shown in Figure 3.

The half space can be flat around the feasible region in many engineering problems, indicating the feasibility to linearize the problem. In the linear model, every point in the feasible space must be within one side of any straight boundary, so the feasible space is convex. Such a convex region can be represented by a circular sequence of vertices $\{v_1, v_2, ..., v_h, v_1\}$. Based on the convexity, the feasible space is assured by connecting the extreme points. In the case of x_i vs. x_j with other parameters fixed, the decision space is solved by a clipping algorithm modified from (Sutherland and Hodgman 1974):

The algorithm works in parallel for each view pane in the decision space. Because each specification *hs* intersects the convex *lp0* at most twice, *m* specifications cost 2*m* intersection calculations at worst. Therefore, it requires $O(m)$ time to solve the local decision space for one view pane.

According to the above algorithm, the problem in Figure 3 is solved again as Figure 4. It is noted that the local decision space is acquired by fixing other design parameters. Thus, the modification of other parameters will change this feasible region. All local decision spaces under different parameter configurations unite the global decision space. Since the global decision space is closely associated with the concept of the performance space, the algorithm will be discussed subsequently.

Given the local decision space, the designer can trivially adjust the design parameters to select feasible and preferred decisions. The benefit of this data visualization, however, is not evident until the facility of the performance attributes from the design modification is provided. As such, the third key component of the representation, the performance space is presented next.

2.3 Performance Space

The local decision space evaluates the specific design vector and assesses the feasible space based on the given specifications. Similarly, the performance space can be visualized as a set of mutual pairings. Let the methodology start with a simple problem with two performance attributes ($y_1 \equiv$ cross sectional area, $y_2 \equiv$ vertical deflection) and two design parameters ($x_1 \equiv$ height, $x_2 \equiv$ middle-thickness). According to the linear performance functions (available in the subsequent case study), the extreme points in the decision space are correspondently mapped to the performance space. Since the linear problem is convex, the feasible performance space is also a convex hull constituted by the mapped extreme points. Utilizing the decision space from Figure 4, the resulting performance space is shown in Figure 5.

The performance space illustrates the feasible region and potential trade-offs between two performance attributes. Suppose that a smaller area and a lesser deflection are preferred. As the design moves from A to B in Figure 5, the area is decreasing while the deflection remains constant. This decision is trivially accepted for every designer. However, the decision to move the performance from B to C is not as agreeable as that from A to B: The deflection decreases as preferred, but the area increases. The same intricacy happens from C to D. Sometimes a weighting coefficient can be set for different performance attributes. For example, an extended weighting method based on the designer's preference could calculate the highest overall expected utility value, given the single utility function and mutually independent relationships (Keeney and Raiffa 1993). However, the independent formation, generation, and validation of multiple performance utilities are difficult. As such, this paper adopts the approach that the designer decides tradeoffs of multiple performance attributes during the interactive design process, aided by the explicit design representation. Provided the decision set, utility functions may not be necessary to discern the preferred design.

The approach is extensible to *m* performance attributes and *n* design parameters. Let the performance attributes $Y^* = \{y_1^*, y_2^*, ..., y_m^*\} \in S$, where S is the total performance space, $S = \{Y \in R^m \mid LSL \leq Y \leq USL, LCL \leq X \leq UCL, Y = f(X)\}$.

Y^* is defined as Pareto optimal (non-inferior) if and only if there does not exist another $Y'=(y_1', y_2', ..., y_m') \in S$, where $Y' \neq Y^*$, such that $y_j' \leq y_j^* \; \forall \; j$ (without loss of generality, the smaller value of the performances is assumed to be better). Therefore, the boundary BCD of Figure 5 is the Pareto Optimal set. Any element in the Pareto Optimal set represents one "optimal" design vector. The term "optimal" here means that there is no way to improve the performance of one attribute without deteriorating other attributes. Therefore, the idea to maximize (or minimize) one single objective function is essentially contradicted with the concept of Pareto Optimal set.

The convex property of the linear problem significantly simplifies the solution of the feasible space. Based on the convexity, the decision space and the performance space are the convex hulls of the same extreme points in two different spaces. Therefore, the first critical step is to find these extreme points. This can be done by solving the system equations composed of n design constraints. Every combination of n constraints from the specification and the parameter limits corresponds to a potential extreme point. The confirmation of this intersection point comes from the feasibility validation of the solution. Any valid intersection point of n constraints is one extreme point of the feasible design space. After all extreme points are acquired, a convex hull algorithm can be applied to each viewing-pane in the decision space and the performance space. Alternatively, the extreme points can be traced to find the facet of the feasible polytope. Each facet represents one specification or parameter limit. It is worth noting that the linear system equations $\boldsymbol{F} \cdot \boldsymbol{X} = \boldsymbol{Y}$ are solved by the LU decomposition method (Nazareth 1987). Given the fact that there are 2^n system equations sharing the same coefficient matrix $\boldsymbol{F}$ but different vectors $\boldsymbol{Y}$, the LU decomposition gives a distinct reduction in the computation time. An outline of the algorithm is given as:

Like most practical problems, the exploration of all feasible space is a high-order polynomial or NP problem. Assuming that decomposing $\boldsymbol{F}$ to $\boldsymbol{L} \cdot \boldsymbol{U}$ is the major consumer of the computation, the constraint combinations dominate the polynomial order of the total time. However, the LU decomposition adopted in the algorithm has decreased the number of the linear system equations from C_{2n+2m}^{n} to C_{n+m}^{n}. When the dimensionality of the problem is under 10, the performance space can be solved in a few minutes using a typical PC. For the following case study, only fractions of a second are required. Moreover, this computation is performed only once for a given design problem and does not impede the interactive investigation of decision variables.

With the feasible performance space, the sensitivity of each design parameter can be dynamically shown in the 2-dimensional design space. Figure 6 illustrates how the design parameters may affect the performance, where the line AB is determined through the point C as:

$$A = \{(y_1, y_2)\in R^2 \mid y_1 = f_1(UCL_1, \overline{x_1}^C), y_2 = f_2(UCL_1, \overline{x_1}^C)\},$$

$$B = \{(y_1, y_2)\in R^2 \mid y_1 = f_1(LCL_1, \overline{x_1}^C), y_2 = f_2(LCL_1, \overline{x_1}^C)\} \quad (5)$$

Similarly, the design sensitivity of other parameters can be formulated. Combined with the Pareto Optimal concept, the designer can adjust the design parameters to the desirable values. While the performance space gives the information where the design should go, the design sensitivity vector indicates how to get there. It is noted that the performance space in Figure 6, where all four design parameters are flexible, is different from that in Figure 5 in which only two design parameters are changeable. Therefore the performance space of Figure 5 is a subset of the space in Figure 6. The point C represents the current design vector, which is located outside the shadowed feasible performance space. According to the decision variables labeled as 1, 2, 3, and 4, one alternative is to decrease the thickness x_3 in order to be inside the feasible space.

3 CASE STUDY

Beam design has been widely used as a straightforward engineering problem to demonstrate multiattribute design methods (Osyczka 1985; Kunjur and Krishnamurty 1997). A schematic picture of the beam structure and its design parameters are given in Figure 7. Three attributes, cross section area y_1, static deflection y_2, and maximum stress y_3 are served to measure the overall performance. It is assumed that the permissible bending stress of the material, σ, equals 10kN/cm^2, and its Young's Modulus, E, equals $2\cdot10^4$ kN/cm^2.

The performance attributes can be formulated as:

$$y_1 = f_1(X) = 2x_2\, x_4 + x_3\,(x_1 - 2x_4) \quad (6)$$

$$y_2 = f_2(X) = PL^3 / (48\, EI)$$

$$= \frac{165746}{x_3(x_1 - 2x_4)^3 + 2x_2x_4[4x_4^2 + 3x_1(x_1 - 2x_4)]}$$

$$y_3 = \frac{M_1 y_1}{I_1} + \frac{M_2 y_2}{I_2} = \frac{150000x_2}{(x_1 - 2x_4)x_3^3 + 2x_4x_2^3} + \frac{1800000x_1}{x_3(x_1 - 2x_4)^3 + 2x_2 \cdot x_4(4x_4^2 + 3x_1(x_1 - 2x_4))}$$

The geometric constraints are:

40cm ≤ x_1 ≤ 80cm, 30cm ≤ x_2 ≤ 50cm,

0.9cm ≤ x_3 ≤ 5cm, 2cm ≤ x_4 ≤ 5cm.

The specifications are acquired from customer's needs (Otto 1996):

$y_1 \leq 400\text{cm}^2$; $y_2 \leq 0.06\text{cm}$; $y_3 \leq 124\text{Mpa}$.

As such, the performance equations can be linearized around the middle point of the control limits:

$$y_1 = -436.35 + 2.95x_1 + 7x_2 + 53x_3 + 74.1x_4 \ (\text{cm}^2) \quad (7)$$

$$y_2 = 0.265 - 0.002x_1 - 0.001x_2 - 0.0025x_3 - 0.0106x_4 \ (\text{cm})$$

$$y_3 = 191.689 - 0.75x_1 - 1.407x_2 - 1.689x_3 - 10.71x_4 \ (\text{Mpa})$$

With the above functions, the design representation for the beam design is established in Figure 8. The initial design, arbitrarily set at the middle point $X^0=(60, 40, 2.95, 3.50)^T$, is infeasible as indicated by its location outside of the local decision space (identified as the darker regions) shown in Figure 8. It is trivial to acquire a feasible solution by observing the local decision space and adjusting any of the parameters into the feasible region. While there are many options to achieve the feasible, the designer, for the sake of argument, is assumed to decrease the middle thickness x_3 and increase the height x_1. Then design parameters become $X^1=(64, 40, 1.72, 3.50)^T$ in Figure 9. For clarity, Figure 9 only shows the parameter space of the updated representation. Compared with Figure 8, all the design parameters are now inside the updated local decision space, and yet the global decision space has not changed.

Given the feasible design vector, the next step is to improve unsatisfactory performance attributes within the specification. There are basically two strategies to maximize the multiple performance attributes. The compensating trade-off approach allows the higher performing attributes to compensate for lower performing attributes, and the non-compensating trade-off approach intends to improve the weakest performance attribute (Otto and Antonsson 1991). Both approaches can iteratively find their ideal points in the performance space.

Without loss of generality, the case study continues by exploring the compensating trade-off approach. One example is shown in Figure 10. The current design vector $X^*=(80, 39.4, 0.9, 2.57)^T$, acquired after a few steps from the first infeasible vector, gives the performance $Y^*=(313.6, 0.036, 47.2)^T$. The figure reveals that X* is a noninferior solution. Nevertheless, other noninferior solutions also exist according to different preferences on the performance attributes. Also, the design sensitivities in Figure 10 illustrate that the noninferior trade-off between the cross-section area, y_1, and the vertical deflection, y_2, can be obtained by adjusting the width x_2 or the bottom-thickness x_4. For the sake of argument, it is assumed that the designer prefers a smaller area. Directed by the arrows, then the performance space leads design parameters to decrease the width x_2 and the bottom thickness x_4 to $X'=(80, 30, 0.9, 2)^T$ and $Y'=(205.5, 0.052, 66.5)^T$. Similarly, the approach can be applied to each mutual space to acquire the desired overall performance.

It is worth noting that the stress attribute was handled as a design constraint in (Osyczka 1985). The statement is compatible with the representation by ignoring the attribute y_3. However, the case study with the stress y_3 provides the possibility that less loading is preferred for longer fatigue life, though further details of the design strategy will not be discussed here for brevity.

4 DISCUSSION

The representation is a beneficial tool in the interactive design process, with the potential to facilitate other design methodologies by solving the feasible set of designs in both the decision space and performance space. For instance, utility has widely been used to select the highest-ranking solution among the design alternatives. In order to rank multiple aspects of the design alternatives, utility functions can be applied to the performance attributes to quantitatively evaluate the design according to the specifications. However, it is difficult as standard practice to combine all the attributes into one overall function and obtain a single score for each alternative. No matter whether the utility is expressed in multiplicative or other forms, it is always questionable to casually combine the subjective and non-commensurable attributes. Instead of determining the final solution, the described representation explicitly catalogs feasible set of design alternatives and leaves the final decision to the decision maker. As an extension, the performance attributes in the representation can be wrapped with a utility function to establish the utility space. This utility space, composed of the pair-wise comparisons of single attribute utility, provides a panoramic view on the overall utility value. With the assistance of the decision space and the utility space, the designer can interactively make the decision regardless of utility independence of multiple attributes.

The representation is currently incorporated for the deterministic design problem. The ideal design vector is likely to be a non-inferior solution located in the boundary of the performance space in order to maximize the likelihood of performance satisfaction. However, given the probabilistic distributions for the design parameters and functional relations, the performance attributes will exhibit variations. Consider the robust design as a two-step design: 1) bring the mean of the design parameters to the target; 2) minimize the variation of performance attributes at this target (Chen, Allen et al. 1996). Then, robust design can be cast as an interactive process to find the reliable design vector with good performance attributes in the performance-based representation. Without any extension of the current representation, the center point of the feasible region can be considered as a robust solution that is less likely to exceed the specification limits under the unexpected noise. Moreover, the performance-based representation can simultaneously consider the robustness (the relative distance from boundary of the feasible space) and the performance (the absolute value of the performance attributes). As such, the designer can adjust the distance of the design vector from the non-inferior boundary in the performance space to acquire the trade-off between the performance and the robustness, though quantitative evaluation of the robustness requires further theoretical development.

The representation also facilitates the evaluation of different conceptual designs. Conceptual design is one of the most important but one of the least understood stages of the design process. In standard practice, the current subjective evaluation of the design configurations often leads the designer into unnecessary iterations (Goodrich and Rinderle 1997). The representation provides a graphical and quantitative evaluation method which reflects the functional relationship between the design parameter and the performance attributes. While the number and functional relations between design parameters varies with each concept topology, the specified performance attributes are typically identical. By establishing the performance space for each design configuration, the designer can gain the understanding of the important aspects of the design alternatives. Although not all parameters are clear at the stage of conceptual design, the described representation embedded with the major performance attributes and critical design parameters will lead the designer to directly proceed to the best design configuration.

As a new approach to evaluate and assist the design process, the performance-based representation is still undergoing important growth. The algorithm currently solves the feasible performance space only for linear systems. Moreover, the amount of information provided by the representation is a little overwhelming at first. Nevertheless, the concept of solving and visualizing the feasible set of intrinsic attributes is believed to be an important factor of improving the engineering design process. Further efforts are currently addressing the use of second-order and stochastic models as well as the incorporation of the representation with other design methodologies.

5 CLOSURE

A design representation has been developed to aid the designer throughout the design process. Based on the convexity of the linear problem, the decision space is solved by iteratively clipping the hyperplane of the specifications. The performance space is the functional mapping of the extreme points of the decision space. The performance-based representation currently utilizes the pair-wise two-dimensional space to envision the hyper decision space and performance space. Such a representation, demonstrated by the case study, can evaluate the feasibility of the design solution, achieve the feasible solutions and eventually lead the designer to a desired trade-off of multiple attributes. Finally, the potential incorporation with other design methodologies reveals the promising application of the performance-based representation.

ACKNOWLEDGEMENT

This research is supported by National Science Foundation, grant #9702797, through the Division of Design, Manufacture, and Industrial Innovation.

REFERENCES

Akao, Y. (1990). *Quality Function Deployment; Intergraing Customer Requirements Into Product Design*, Booknews, Inc.

Antonsson, E. K. and K. N. Otto (1995). “Imprecision in Engineering Design.” *Journal of Mechanical Design* 117(B): 25-32.

Bras, B. and F. Mistree (1995). “A Compromise Decision Support Problem for Axiomatic and Robust Design.” *Journal of Mechanical Design* 117(1): 10-19.

Chen, W., J. K. Allen, et al. (1996). “A Procedure for Robust Design: Minimizing Variance Caused by Noise Factors and Control Factors.” *Journal of Mechanical Design* 118(4): 478-485.

Cormen, T. H., C. E. Leiserson, et al. (1990). *Introduction to Algorithms*, The MIT Press.

Goodrich, M. D. and J. R. Rinderle (1997). *An Objective Approach to Design Configuration Selection*. Proceedings of ASME Design Engineering Technical Conferences, Sacramento, CA, DETC97/DTM-3866.

Hazelrigg, G. A. (1998). “A Framework for Decision-Based Engineering Design.” *Journal of Mechanical Design* 120(4): 653-658.

Iwasaki, Y. and e. al. (1993). *How Things Are Intended to Work: Capturing Functional Knowledge in Device Design*. Proc. Int'l Joint Conf. AI, Menlo Park, California, AAAI Press.

Keeney, R. L. and H. Raiffa (1993). *Decisions with multiple objectives : preferences and value tradeoffs*. Cambridge [England] ; New York, NY, USA, Cambridge University Press.

Kunjur, A. and S. Krishnamurty (1997). “A Robust Multicriteria Optimization Approach.” *Mechanism and Machine Theory* 32(7): 797 - 810.

Kusiak, A., J. Wang, et al. (1996). “Negotiation in Constraint-Based Design.” *Journal of Mechanical Design* 118(4): 470-477.

Nazareth, J. L. (1987). *Computer Solutions of Linear Programs*, Oxford University Press.

Osyczka, A. (1985). Multicriteria Optimization for Engineering Design. *Design Optimization*. J. S. Gero, Academic Press, Inc.**:** 298.

Otto, K. N. and E. K. Antonsson (1991). “Trade-off Strategies in Engineering Design.” *Research in Egnineering Design* 3(2): 87-104.

Paz-Soldan, J. P. and J. R. Rinderle (1989). *The Alternate Use of Abstraction and Refinement in Conceptual Mechanical Design*. ASME Winter Annual Meeting, San Francisco, CA.

Rinderle, J. R. and V. Krishnan (1990). *Constraint Reasoning in Concurrent Design*. ASME, Design Technical Conference on Design Theory and Methodology, Chicago, IL.

Stone, R. B. and K. L. Wood (1999). *Development of A Functional Basis for Design*. Proceedings of ASME Design Engineering Technical Conference, Las Vegas, Nevada, DETC99/DTM-8765.

Sturges, R. H., K. O'Shaughnessy, et al. (1996). “Computational model for conceptual design based on extended function logic.” *Artificial Intelligence for Engineering Design, Analysis and Manufacturing: AIEDAM* 10(4): 255-274.

Suh, N. P. (1990). *The Principles of Design*, Oxford University Press, Inc.

Sutherland, I. E. and G. W. Hodgman (1974). “Reentrant polygon clipping.” *Comminications of the ACM* 17(1): 32-42.

Szykman, S., J. W. Racz, et al. (1999). *The Representation of Function in Computer-Based Design*. Proceedings of ASME Design Engineering Technical Conference, Las Vegas, Nevada, DETC99/DTM-8742.

Taguchi, G. and S. Konishi (1992). *Taguchi Methods, Research and Development*. Dearborn, Mich., ASI Press.

Thornton, A. C. (1996). “The Use of Constraint-Based Design Knowledge to Improve the Search for Feasible Designs.” *Engineering Applications of Artificial* 9(4): 393-402.

Thurston, D. L. (1991). “A Formal Method for Subjective Design Evaluation with Multiple Attributes.” *Research in Engineering Design* 3(2): 105-122.

Umeda, Y. and T. Tomiyama (1997). “Functional Reasoning in Design.” *IEEE Expert* 12(2): 42-48.

LIST OF TABLES AND FIGURES

Table 1. Local Decision Space Solution

1. Initialize the loop $lp1 = \{(LCL_i, LCL_j), (LCL_i, UCL_j), (UCL_i, UCL_j), (UCL_i, LCL_j), (LCL_i, LCL_j)\}$, and $k = 1$;
2. Create the half space hs defined by the specification k, $hs = \{X \in R^n \mid f_k(X) \geq LSL_k\}$. For two-side specifications, repeat Step 3~5 for USL_k;
3. Set $lp0 = lp1$, $v_a = v_1$, and $v_b = v_2$. Empty $lp1$;
4. If both v_a and v_b are inside hs then

    Add v_b into $lp1$

   Else if one of v_a and v_b is inside hs then

    Calculate the intersection point v_c of $v_a v_b$ and hs

    Add v_c into the new loop $lp1$

    If v_b is inside hs then

     Add v_b into $lp1$
5. Set $v_a = v_{a+1}$ and $v_b = v_{b+1}$. If not all vertices are done, go to step 4;
6. $k = k + 1$. If $k \leq n$, go to step 2;
7. Add the first vertex into the end of $lp1$ to finish the loop. The region surrounded by $lp1$ is the feasible space of x_i vs. x_j with other parameters fixed.

Table 2. Global Performance Space Solution

1. Choose n distinctive constraints from all m specifications and n parameter limits;
2. LU decompose $\boldsymbol{F}$ into $\boldsymbol{L}\cdot\boldsymbol{U}$;
3. Constitute one distinctive right-side $\boldsymbol{Y}$ from specifications and parameter limits;
4. Solve $\boldsymbol{L}\cdot\boldsymbol{U}\cdot\boldsymbol{X} = \boldsymbol{Y}$ by forward and back substitution. If the solution X^* satisfies all the specifications, add it into the extreme point list.
5. If all distinct $\boldsymbol{Y}$s have been operated, go to next step. Otherwise go to step 3;
6. If all distinct constraint combinations have been operated, go to next step. Otherwise go to step 1.
7. All extreme points and the facets are now available. The feasible decision space and performance space can be acquired by projecting the facets into each 2-dimensional view pane, though other projections and representations may be beneficial.
8. The boundary of the feasible space in each view pane is a convex hull of the extreme points. Graham-scan algorithm can be adopted to retrieve the convex boundary in $O(h\lg(h))$ time where h is the number of all vertices (Cormen, Leiserson et al. 1990).

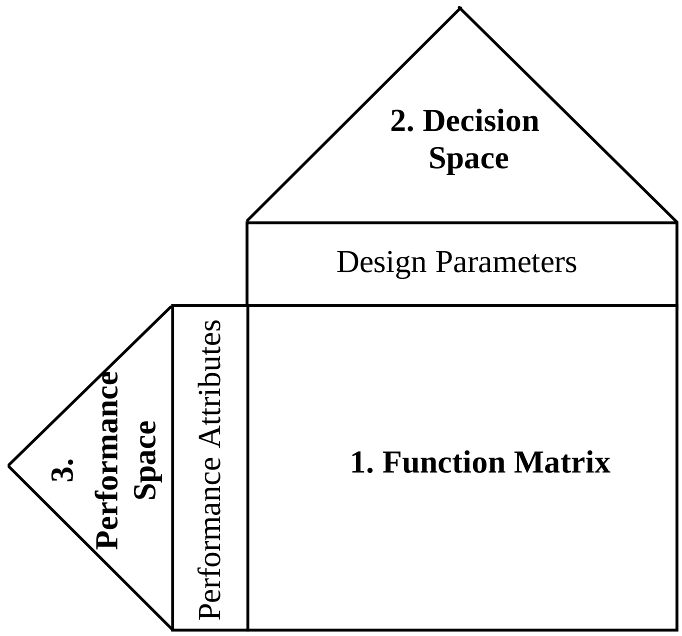

Figure 1. Performance Orientation Chart

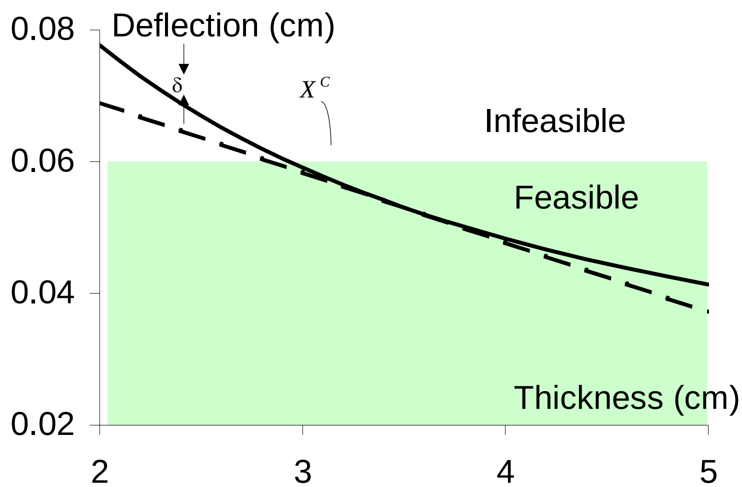

Figure 2. Performance Function and Specification

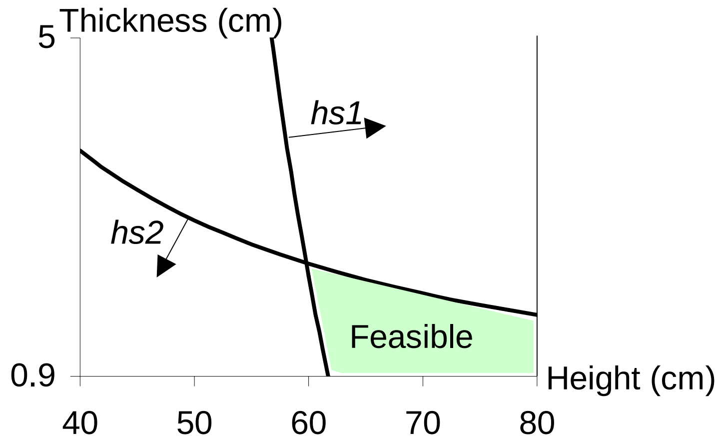

Figure 3. Local Decision space of x_1 vs. x_2 with Other Parameters Fixed

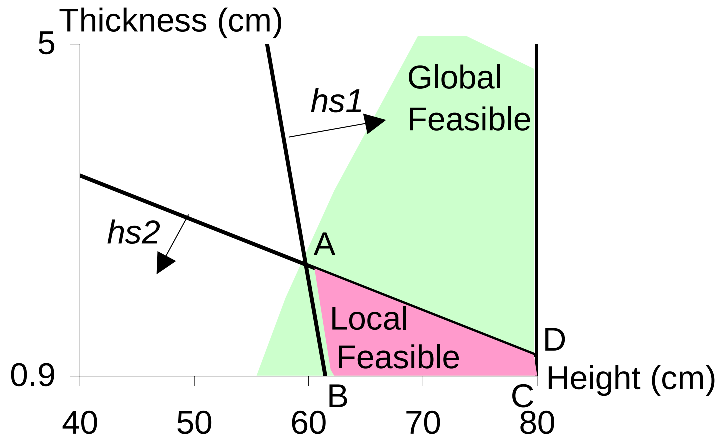

Figure 4. Decision space of x_1 vs. x_2 after linearization

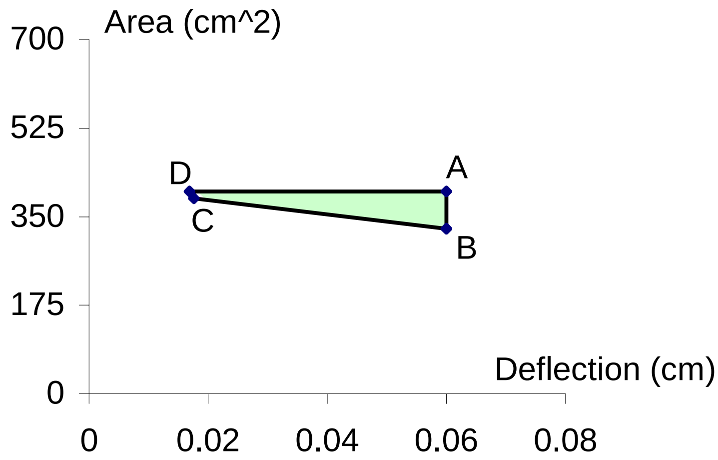

Figure 5. Performance Space of y_1 vs. y_2

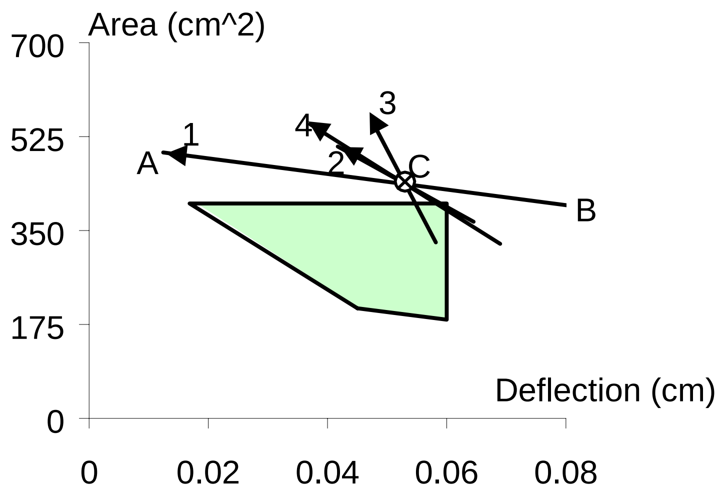

Figure 6. Design Sensitivity and Performance Space

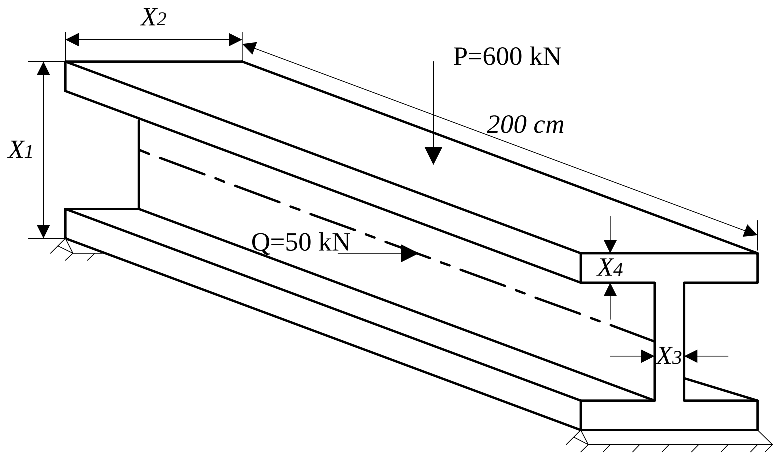

Figure 7. Beam Design

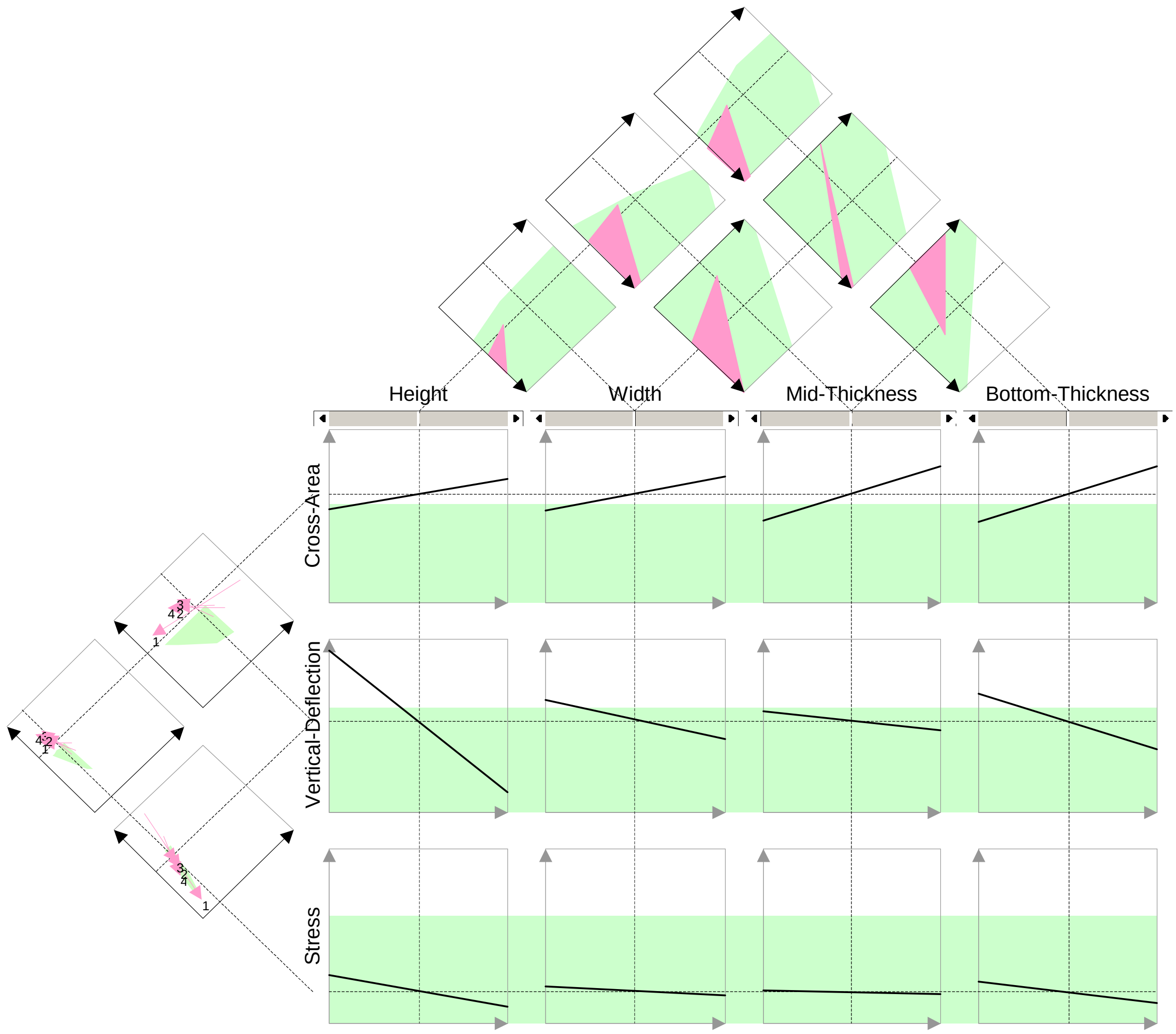

Figure 8. Performance-based Representation with Beam Design

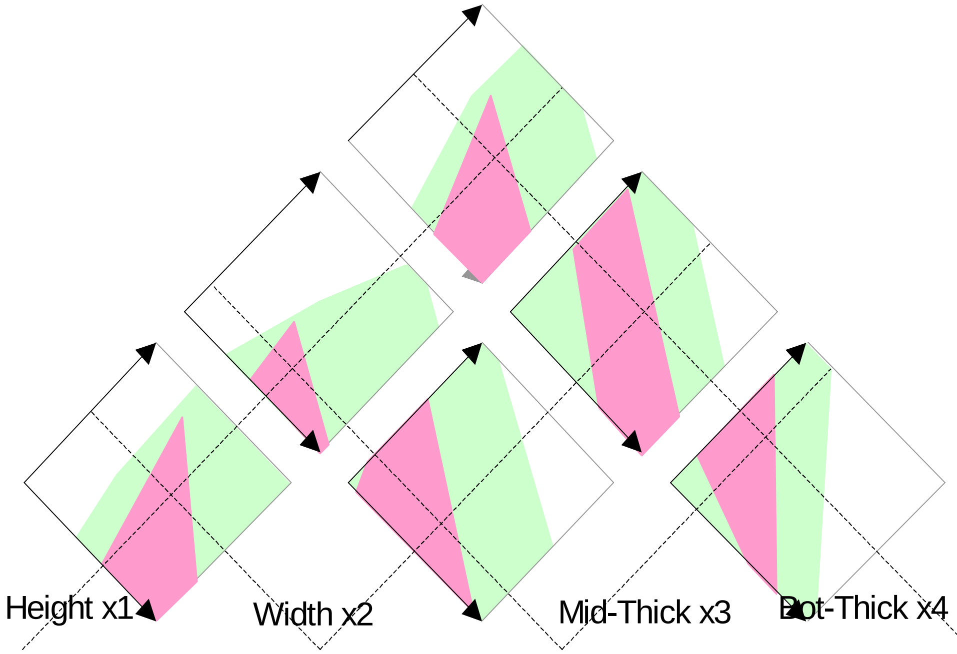

Figure 9. Feasible Design

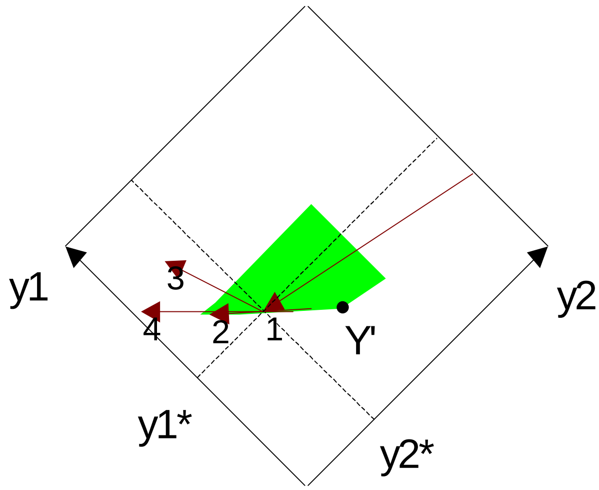

Figure 10. Trade-off Design