A Spectral Optimization Algorithm for Multi-Objective Prototype Selection

Andy Dong
Graduate Student Researcher
Berkeley Expert Systems Technology Laboratory
6102 Etcheverry Hall
Berkeley, CA 94720
(510) 643-8146
adong@jerry.Berkeley.EDU

Alice M Agogino
Professor of Mechanical Engineering
University of California at Berkeley
5136 Etcheverry Hall
Berkeley, CA 94720
(510) 642-6450
aagogino@euler.me.Berkeley.EDU

Abstract

In design synthesis, engineering prototypes make an ideal representation medium for preliminary designs. Unlike parametric design wherein a pre-specified design is parametrically varied, design synthesis demands artistic creativity and engineering experience to transform the previously known components, relationships and designs into a new form. The process compels the designer to ascertain which prototypes will, in some sense, best satisfy the design task. The challenge in this assignment lies in selecting the "right" design prototype. This selection process typically entails an objective evaluation of different designs that perform the same functions or have similar intended behavior and comparing trade-offs between alternate designs. This paper introduces a multi-objective spectral optimization algorithm for the selection of design prototypes based upon their functional representations. The optimization algorithm returns an index of rank, scoring the functional similarity of the proposed design to the goal design. Two illustrative examples apply the algorithm to the selection of a heat fin and beam.

Keywords: design methods and models; computational methods of design; prototype selection

1 Introduction

1.1 The Process of Prototype Selection

Gero et. al. [10] define a prototype as “a conceptual schema for the representation of generalized design knowledge” that unifies an intended interpretation, a vocabulary of design elements and knowledge. For example, a prototype representing a rectangular beam captures the commonality of all beams with an infinite variety of possible cross sections, lengths, materials, and so on. From each prototype, an associated parametrized design description can be developed. At any point in this process, the designer may determine that it will not be possible to refine the original prototype into an acceptable product and new prototypes must be generated through some innovation process. The final parametrized design is then used in the detailed design phase to make commitments to specific design variables. [17]

Selection of the prototype requires a synthesis of experiential knowledge and technical skills to consider how the behavior of each prototype might satisfy the design objectives. For this reason, the task of prototype selection has been treated in a multitude of ways in the literature, ranging from simple rankings of the candidates by a weighted sum of attributes to formal multi-objective optimization methods. In this first section of the paper, we discuss relevant methods to motivate the development of a method for prototype selection based on spectral analysis from signal processing. The following section describes the spectral optimization approach to prototype selection and its impact on design methodology. The remainder of the paper discusses two illustrative examples and draws conclusions.

1.2 Heuristic Approaches to Prototype Selection

Much research has been devoted to the subject of selecting design alternatives. Since prototype selection is often a domain dependent problem, several approaches use heuristics to develop a computational strategy that best utilizes the available resources. Two quantitative measurements of qualitative judgments related to the approach described in the paper are discussed. First, Ishii [11] has proposed a Design Compatibility Analysis (DCA). The analysis computes a match index between design requirements and the description of the proposed design. Each design element *s* is represented as an object which has a design value as its attribute. The match index sums up the match coefficient *M(s)* for each design element *u(s)* multiplied by its corresponding weighted importance through the equation $MI_i = \sum u(s) \times M(s)$.

A systematic method proposed by Iyengar [13] constructs a computation network composed of design constraints and the designer preferences. For each alternative design, the computation network computes, propagates, and consolidates "goodness-of-match" indices for each of the functional characteristics and selects the "best" component. The indices were often based on expert knowledge rather than objective functions. However, the method is useful for systems evaluations for which a functional representation would be difficult to derive.

In summary, like the proposed spectral algorithm, these methods specify a metric for measuring the deviation of the performance characteristics of a design prototype from the design objective. These methods look at the simplest version of prototype selection, the selection of a single element. This contrasts with the configuration and selection of multiple elements as treated by Ward and Seering [21].

1.3 Prototype Selection as a Multi-Objective Selection Problem

The evaluation of prototypes often requires balancing multiple performance characteristics. To perform such a trade-off analysis on multiple objectives, much research has treated prototype selection as a multi-objective optimization problem. Since the method proposed is quantitative, multi-objective design applying qualitative reasoning is not discussed. [2] The formal quantitative methods discussed below have been pursued in the literature for multi-objective selection in mechanical design. [3] [4] [15]

The multi-objective decision-making process consists of a sequence of decision and computation phases. In the decision phase, the decision-maker decides whether or not a solution is optimal with respect to his preferences. The algorithms use the information in the preferences to determine the direction in which one expects to obtain a better solution. In the computation phase, the new solution is generated for the next decision phase. The procedure is stopped when an optimal solution which reflects the decision-maker's preferences is found. Such a sequence of operations results in a selection of Pareto-optimal solutions which represent a subset of available solutions. [7]

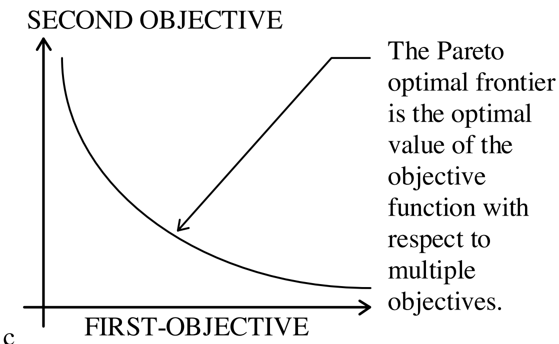
Several researchers have developed multi-objective decision-making algorithms. A standard approach is to vary parametrically the weighting coefficients **w** on the objectives, finding all non-dominated solutions. This problem is formulated as:

$$\min \mathbf{x} \{ \mathbf{w}^T f(\mathbf{x}) : g(\mathbf{x}) \le 0 \} , \mathbf{w} \ge 0 , \Sigma w_j = 1 .$$

where f(**x**) represents the preference function, g(**x**) denotes the constraints and j represents the j-th objective function. By varying the weighting coefficients w_j, the method yields all solutions of the functional-efficient boundary. The iteration is terminated when the breaking-off criterion,

$$\left| f_j(\mathbf{x}) - \mathbf{y}_j \right| < \varepsilon \ \forall \ j = 1 \dots m \text{ with } \in > 0 .$$

where $\mathbf{y}_i$ is the i-th objective, is satisfied. [7] The points on the efficient frontier of optimal solution parameters with the proper weighting factors are called the Pareto optimal set.



c

Another algorithm called the STEP-method [20] assumes that the preferred compromise solution should be as close as possible to the ideal objective function vector with an upper bound tolerance of **y**. For this method a scalar substitute problem is applied by solving a non-linear minimization problem with bounded and weighted objectives. If some of the solutions of a compromise solution are satisfactory and others are not, the decision-maker must indicate a certain amount of relaxation of at least one satisfactory objective to improve the unsatisfactory ones in the next iteration. Otherwise, the preferred solution is determined by the compromise solution. In most cases, more than one functional-efficient point $\mathbf{Y}^0$ is determined although the approach by Jahn steps across the interior of the criterion space **Y** in the direction of the preferred solution and finds only one $\mathbf{Y}^0$.

One method proposed by Bradley and Agogino is called Intelligent Real Time Design (IRTD). In the IRTD approach, the goal is to focus the designer's limited resources on searching those portions of the design space which yield information most relevant to selecting the best design, and eliminating or reducing uncertainty where it has the greatest impact on the utility of the final design. IRTD was formulated for prototype selection problems [5] and has recently been successfully applied to "Catalog Design" wherein the designer selects design components directly from a catalog. [6] This method relies on the expected utility of a candidate prototype to select design options. The design objectives describe a utility maximizing function, which might contain designer preferences, that will result from selection of that prototype. The weakness of IRTD lies in the difficulty of constructing a complete and accurate utility function and in reducing the complex expressions of utility to a tractable form by making appropriate assumptions and simplifications.

In the design selection problem discussed by Mistree [16], the authors present a linear programming model to solve constrained, multi-objective optimization problems for preliminary selection. Mistree defines preliminary selection as the portion of the design stage wherein the designer identifies the principal criteria influencing selection and the relative importance of the criteria based upon soft information. Then, the designer ranks the concepts in order of preference based on multiple criteria and their relative importance. The authors present a decision-support problem (DSP) to assist the designer in the concept selection based upon the results of the linear programming solution. This method is more interactive than the proposed spectral algorithm but is computationally expensive for large problems since the DSP must locate the associated design vector associated with each alternative that satisfies the constraints. Also, the DSP often treats attributes as scalar values rather than, for example, a function of the design variables. Though this reduces the non-linearity of the problem, it may present some difficulty in handling non-linear objectives and constraints.

From a multi-objective perspective, the proposed spectral optimization approach treats each functional characteristic of the prototype as a “vector component” in the space of design objectives. For example, if the criteria for beam selection were strength, weight and stiffness, each of these criteria would be represented as a vector direction. However, rather than modifying the design vector or the preference coefficients for each criteria to arrive at a functional-efficient point, the algorithm measures the “goodness-of-fit” by analyzing the functional compatibility of each respective prototype to the design goal over the local variable space. This analysis is possible by partitioning the criteria into vector components and then applying a technique known as spectral optimization. The algorithm will show how the respective vector components of each prototype correlates to the spectral energy content of the particular prototype. This energy content, we argue, quantifies the "goodness-of-fit" of the prototype to the design objectives.

1.4 Spectral Optimization for Prototype Selection

What motivates the current work is that the methods discussed above appear to fall short in one of the following goals of prototype selection algorithms. On a practical level, we wish to avoid arbitrary quanitification of qualitative factors based upon heuristic rules or expert systems analysis. Furthermore, the algorithm should be based upon readily identifiable functional parameters rather than qualitative judgments or preference functions which are difficult to formulate. Second, for prototype selection, the exact value of the design vector associated with the prototype that satisfies the multi-criteria objectives is not important. Rather, prototype selection concerns with measuring the prototype's deviation from the desired functionality over the range of feasible design variables and relative comparison between competing prototypes. Thus, a full multi-objective Pareto optimality analysis on each prototype, a computationally expensive operation, may not be necessary. Finally, prototype selection must consider relative comparisons between competing concepts over the feasible design space and the functional space.

For prototype selection, the interest is not necessarily *how* the variables affect the objectives but also *how far away* from the objective is the prototype. The spectral approach examines each prototype over the design variables space and then returns a measurement of the proximity of the prototype to the ideal point in the magnitude of the spectral weighting coefficient. This measurement informs the designer of the dominant prototype through the value of the weighting coefficients. Knowing the dominant prototype, the designer could then apply multi-objective selection techniques to locate the optimal design vector associated with the selected prototype.

The heuristic approaches do not meet the practicality requirement outlined. Also, the heuristic approaches are domain dependent and limited to application where the expertise is available to provide the needed heuristics and the decision-making context is well-defined. The solution report of the multi-objective selection algorithms cited also does not meet these requirements. First, the multi-objective techniques often specify gradient directions for moving the design variables such that the resulting solution progresses towards the multi-objective solution point. In essence, these methods modify the vector of design variables to reach a compromise solution by modifying the preference weighting on each attribute. An extension to prototype selection is awkward and indirect in that it deals with the variation of a single design vector associated with a prototype rather the deviation of the prototype's behavior from the desired over the range of design vectors. Thus, although the multi-objective algorithms offer pairwise dominance information, (comparison of prototype A to prototype B for two objectives) the multi-objective are difficult to use as the number of objectives are larger than two.

The spectral optimization algorithm introduced in this paper satisfies the criteria stated above. First, the spectral algorithm considers n prototypes over the local space of the design variables simultaneously. Second, the spectral method offers a uniform, quantitative measurement to compare the prototypes to a single goal for relative comparison. Finally, the algorithm is based upon engineering equations derived from the prototype geometry or for which the relevant performance measurements appear explicitly as a function of some design variables. The goal of the proposed algorithm is to inform the designer of the "best" design prototype which satisfies a prescribed function in a quantitative and rigorous manner. The algorithm comes from an optimization technique in linear algebra which calculates the optimal approximation of a function as the orthogonal projection of its components in n-space. This method is applied in signal processing theory as spectral optimization to obtain the Fourier series of a signal. Here, the principal is that the coefficients of the Fourier series correlate to the dominant energy frequencies. We describe how to extend this principle by showing that these coefficients indicate the dominant prototype.

2 Spectral Optimization Algorithm

2.1 Description

The objective of the algorithm is to offer a quantitative measurement of the relative merit of alternative designs based upon various functional representations. We look at the suitability of the design over the local space of the design variables. Interpretation of the results is based upon signal processing techniques.

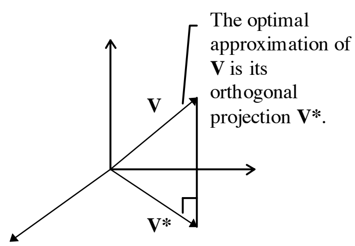
First, the designer selects a set of suitable design solutions for the intended objective. Let **b** = $\{b_1, b_2, b_3, ..., b_k\}$ a linearly independent set of design solutions drawn from the hyperspace of all possible designs **R**. Furthermore, define **b** as an inner product subspace which spans **R** (Hilbert Space). By linear independence, we mean that no design could be constructed through standard Boolean operations on the other designs, such as union, intersection or difference.

Next, a vector of functional parameters is derived. Let **V** denote the desired (perhaps optimal) design. The **V** is a vector-valued function which captures the engineering role and design attributes of the desired design.

From linear algebra [1], the optimal approximation of **V** in the subspace **b**, denoted by **V** *, is given as a linear combination of the b_i.

$$V^* = \sum_{i=1}^{k} \alpha_i b_i \qquad \textbf{Equation 1}$$

This is known as the **Optimal Approximation Theory** and has been applied to curve fitting algorithms and Fourier analysis. A simplified proof of the Optimal Approximation Theory is given in the Appendix. Essentially, this theory states that the optimal approximation of a function on a subspace is its orthogonal projection.



The α_i determine the mapping coefficients. From linear algebra, the α_i are given by the transition matrix

$$\begin{bmatrix} \alpha_1 \\ \vdots \\ \alpha_k \end{bmatrix} = \begin{bmatrix} < b_1, b_1 > & \cdots & < b_1, b_k > \\ \vdots & \ddots & \vdots \\ < b_k, b_1 > & \cdots & < b_k, b_k > \end{bmatrix}^{-1} \begin{bmatrix} < V, b_1 > \\ \vdots \\ < V, b_k > \end{bmatrix}$$

Equation 2 Transition Matrix

In this context, the vector **V** * does not signify that the optimal solution will be composed of α_i β_i elements. Rather, the question that these parameters answer is: *Given a set of feasible design solutions, which design is "most-likely-to-succeed" under the design objectives?* Mathematically stated, "*Which design has the smallest distance from the desired objective normalized over the local space of the design variables*?"

To answer the question why the magnitude of the weighting parameter α_i identifies the relative superiority of one design over another, we appeal to signal processing theory. The significance of the α_i is analogous to spectral analysis in signal processing theory. In signal processing, one is concerned with determining the dominant frequency components of a signal. A signal reconstruction technique such as the Fourier transform could be derived using the *Optimal Approximation Theory*. Using this technique, the α represent the energy content of a signal at a particular frequency. [23] The higher the magnitude of

α, the more information the signal carries at that frequency. The magnitude of the α correlates with the "dominant frequency." Completing the analogy to the design problem, the vector of desired objectives maps to the signal and the feasible design solutions map to the frequency components. Thus, a design element multiplied by a large α will signify a better match between design prototype and design goals.

We can mathematically restate the above conclusions as follows. The geometry of a design prototype associates a set of functional attributes to that component, for example $P_1 = p_1(\mathbf{x}), p_2(\mathbf{x}) \ldots p_n(\mathbf{x})$ where P is the particular prototype, p_i is a single function and $\mathbf{x}$ is the vector of design variables. One can potentially characterize all objectives which describe the final goal design, for example $g_1(\mathbf{x}), g_2(\mathbf{x}) \ldots g_n(\mathbf{x})$ where each g_i is a function. By mapping the desired design solution to the sub-space of feasible design alternatives through the spectral optimization algorithm, we are given the set of spectral parameters α for all prototypes P_i. The parameters specify the design concept which best satisfies the objectives. Optimization studies could then be concentrated on the chosen design with confidence that a better solution will not lay in the design alternatives omitted.

2.2 Comparison to Goal Programming

The algorithm suggested may be likened to goal programming algorithms. A solution using goal programming represents one single solution on the efficient Pareto-optimal frontier. Many of the goal programming methods specify some goal vector $\mathbf{g}$, that is, a goal g_i is set for each of the N criteria. Since not all goals are attainable, one then provides a measure of deviation from the specified goal, with the aim of minimizing the deviation. The measure of closeness of the solution $\mathbf{y}$ in R^N is generally accomplished by metrics. A common measure of the deviation is the L_p metric [20]

$$G(y) = \left[\sum_{i=1}^{N} (y_i - g_i)\right]^{1/p}$$

Suppose that c_i^- and c_i^+ denote the under achievement and the over achievement of the *i*th goal. The following programming problem minimizes the deviation from the specified goal:

$$\text{minimize} \sum_{i=1}^{N} (c_i^- + c_i^+)^{1/p}, 1 < p < \infty$$

The achievement functions $h_i(c^+,c^-)$ are linear in the achievement variables c^+ and c^-. These achievement functions are minimized in sequence in such a way that a lower ranked achievement function cannot be optimized to the detriment of a higher ranking one.

In the proposed methodology, we measure the deviation from the goal through the magnitude of the spectral coefficients. As shown in the proof, the optimal approximation is achieved when no other point in the design space lies closer to the desired solution point than the one found. In other words, we have minimized the over and under achievement. Also, the solution reveals the dominance of the vector approximations in the magnitude of their coefficients. These coefficients were used to determine dominant prototypes. The algorithm is more appropriate than general goal programming for prototype selection since we get relative comparisons directly. Also, goal programming locates the particular set of design variables associated with a prototype that minimizes the over and under achievement rather than reporting how well the prototype potentially meets the design objectives over the space of feasible design variables. To correlate the proposed algorithm with other goal oriented algorithms and approximation theorems, the algorithm may be summarized as: We ask for an appropriate polynomial $\mathbf{p}$ that approximates $\mathbf{f}$. The polynomial of $\mathbf{p}$ is composed of vector valued functions describing each of the possible prototypes. The polynomial $\mathbf{f}$ describes the desired functionality. The coefficients of the polynomial $\mathbf{p}$ denote the dominant prototypes. Figure illustrates the role of the spectral algorithm in a prototype selection methodology.

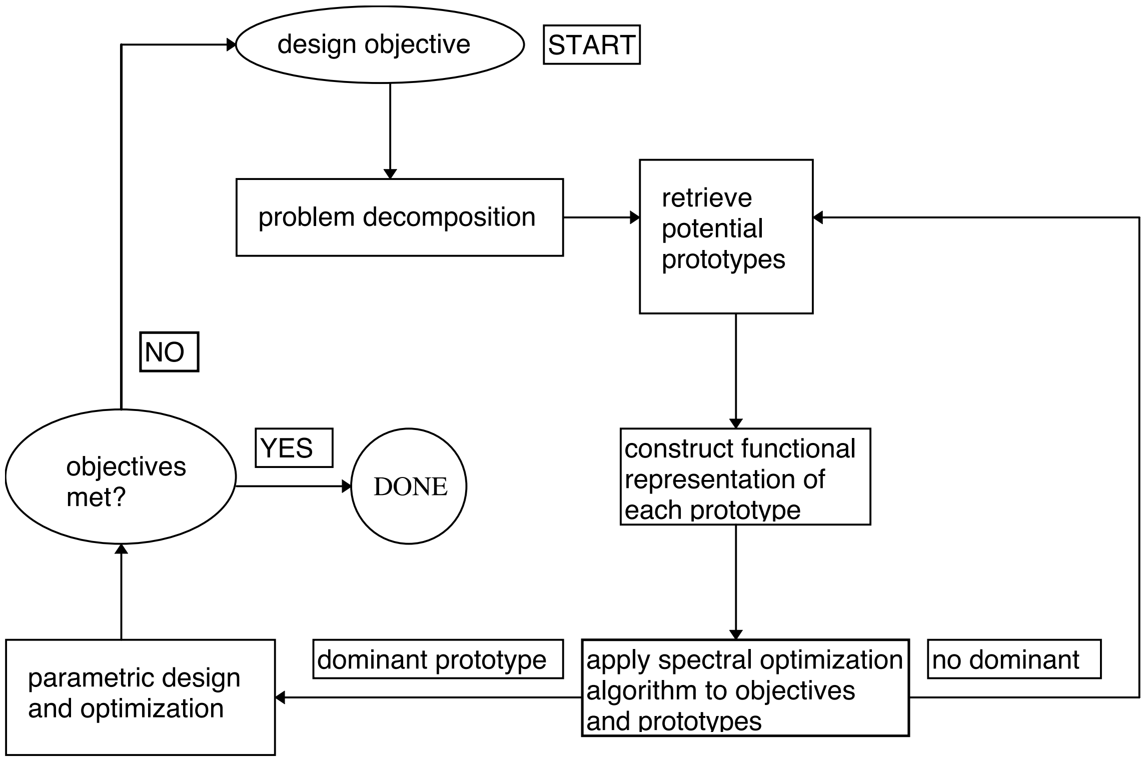


Figure 1 Spectral Optimization in the Prototype Selection Process

3 Examples

To test the theory proposed, we apply the algorithm to the selection of a heat fin and a beam. The results correlate with expected results.

3.1 Heat Fin

The design of heat dissipation fins on a flat plate is considered. The three design alternatives chosen were the rectangular fin, the triangular fin and the parabolic fin. Therefore, the set of b_i's is given by

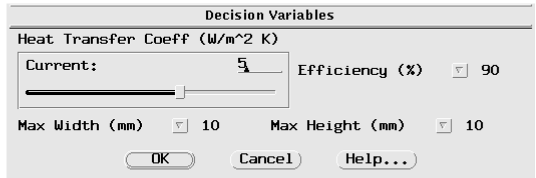
$$b = \begin{bmatrix} \text{rectangular fin} \\ \text{triangular fin} \\ \text{parabolic fin} \end{bmatrix}$$

Two criteria may be used to describe the functionality of the fin. The parameters map a functional representation to the fin design.

1. Heat flow efficiency - a volumetric efficiency based upon the heat transfer for a given volume of material
2. Heat transfer coefficient

For extended surface heat dissipation devices, the heat flow efficiency and heat transfer coefficient can be found in the literature. [11] [14]

Suppose that it is desired that the fins have an efficiency of 90%, a heat transfer coefficient of 0.75 W/m K, a maximum height of 50 mm and width of 10 mm. An AutoCAD™ ADS application has been written to prompt the designer for functional criteria and topological constraints (integration boundaries). A sample dialog from the application is shown below.



Following the computation, the fin is drawn into the CAD session. The computation of the algorithm resulted in a coefficient matrix of

$$\alpha = \begin{bmatrix} 0.163 \\ 23.983 \\ -24.003 \end{bmatrix}$$

The magnitude of the coefficients weight the relative suitability of the design feature. The sign of the coefficient is not relevant. Thus, one should choose the parabolic fin over the rectangular and the triangular fin. Calculating the heat transfer coefficient for the rectangular fin using the maximum limits of the integration variables revealed that the heat transfer coefficient was 0.71 W/m^2 K, below the threshold value. For the same dimensions, the heat transfer coefficient for the triangular and parabolic fin were $h_t = h_p = 0.78$ W/m^2 K. The respective fin cross-sectional area were $A_t = 2.5(10^{-4})$ m^2 and $A_p = 1.7(10^{-4})$ m^2. Since $A_p < A_t$, this indicates that the parabolic fin had a slightly better volumetric efficiency than the triangular fin.

These results agree with fin heat transfer studies performed by Incropera and DeWitt. According to Incropera and DeWitt [11], a straight triangular fin is attractive because, for equivalent heat dissipation, it requires much less fin material than the rectangular fin. In this regard, heat dissipation per unit volume is largest for a parabolic profile. However, it is only slightly larger than that for a triangular fin. The small difference between the coefficient corresponding to the triangular and parabolic fins confirms this statement. If a manufacturability function were constructed based upon the design geometry, perhaps the suitability of the rectangular fin would dominate.

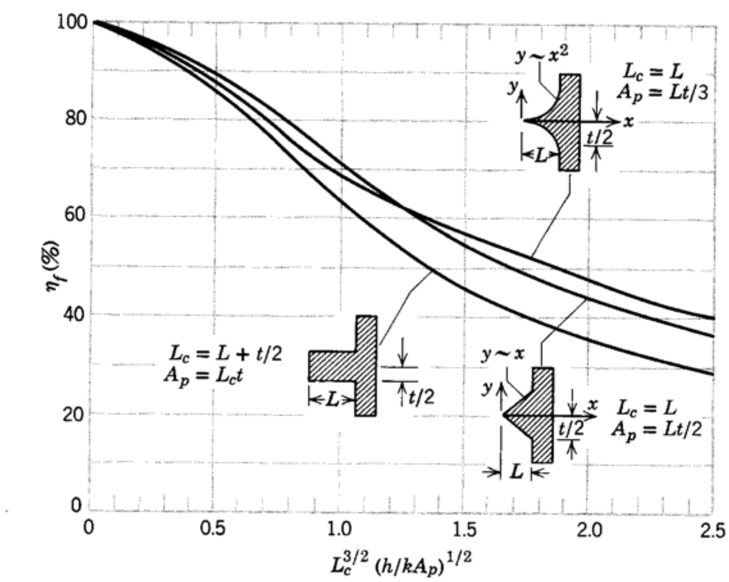


Figure 3.18 Efficiency of straight fins (rectangular, triangular, and parabolic profiles).

The above figure [11] depicts that the parabolic fin generally has better volumetric efficiency than the triangular fin. For our design case, where the value of abscissa is on the order of 0.2, the

parabolic and triangular fin have essentially identical efficiencies.
We do not consider the rectangular fin since its heat transfer coefficient is below the threshold value.

3.2 Beam in Bending

Similarly, one can apply the algorithm to the classic beam-in-bending problem under a concentrated load wherein one is interested in selecting the right beam geometry based upon stress, deflection and weight. For example, suppose we wish to choose among the following beam cross sections:

$$b = \begin{bmatrix} \text{rectangular} \\ \text{hollow box} \\ \text{I - beam} \end{bmatrix}$$

The three parameters are (normalized by the load, beam length and Young's Modulus)

stress parameter	$\sigma = I/c$
deflection parameter	$\delta = I$
weight parameter	$\rho = A$

where I is the moment of inertia, c is the distance from the neutral axis to the outer fiber, and A is the cross section area. Suppose for some application that the desired deflection is no greater than $\Delta = 2.5$ mm when a 5000 N force F is applied at the center of a 5 m (L) beam. The yield stress of AISI 1040 HR steel is $S_y = 300$ MPa. Young's Modulus is 207 GPa. The density of steel is 7854 kg/m^3. Computing the transition matrix over a feasible range of beam cross-section dimensions resulted in a selection matrix of

$$\alpha = \begin{bmatrix} 5.044 \\ -8.195 \\ 8.892 \end{bmatrix}$$

favoring the I-Beam.

To verify the results, the weight parameters were minimized using the optimization package GINO™. Three separate optimization problems were run. The objective is to minimize the cross-sectional area. The dimensions of the beam cross-section must satisfy the dimensional constraints to satisfy the stress and deflection requirements. Minimizing the cross-sectional area assures minimum weight since the weight is scaled by the length of the beam and the density of the beam material. The optimization problem was formulated as

$$\begin{aligned} &\min \textit{imize area} \\ &\text{such that} \\ &\frac{I}{c} \ge \left[\frac{I}{c}\right]_{des} \quad \text{stress constraint} \\ &I \ge I_{des} \quad \text{deflection constraint} \end{aligned}$$

The results indicated that the solid rectangular beam would have a mass of 190 kg, the hollow box beam would have a mass of 126 kg and the wide-flange beam would have a mass of 97.4 kg. This result corroborates the values of the coefficients obtained. The wide-flange I-beam best satisfies the desired objectives.

4 Summary

This paper discussed a methodology for mapping a desired design solution to a sub-space of feasible design alternatives using a spectral optimization algorithm. The method involves applying the Optimal Approximation Theory to map the space of feasible design solutions to the space of the "desired" design solution to determine which of the feasible solutions is dominant. Each of the design objectives is represented by a vector component. The vector components contribute to the “spectral energy” of the prototype design. The spectral optimization algorithm computes an index which ranks the “goodness of fit” of the design prototypes based upon its “spectral energy” content. Using the information garnered from the mapping process, the designer is equipped with a quantitative evaluation of which design geometry will potentially best satisfy the design criteria.

The algorithm is applicable to prototype selection for which there are relatively few prototypes to choose over (perhaps less than five) and mathematical representations of the prototype behavior based upon the design variables exist. It is particularly useful in the field of feature-based design since the mathematical functionality of the design is explicitly derivable from the geometry of the form-features. [8] [9] Shah [19] defines features as *entities which represent the engineering meaning of the geometry of a part or assembly*. The feature is a physical constituent of a part which encapsulates the engineering significance of portions of the geometry. Additionally, it might be possible to extend the algorithm to manufacturability analysis since feature-based manufacturability analysis is also based upon the geometry of the form-features. Although the examples use constant objectives, it is possible to measure deviation based upon functional objectives of the design variables.

In particular, the algorithm has application to an ongoing project to develop a conceptual design information server. [22] One issue is reducing the number of conceptual designs retrieved to one or two concepts with which to customize. Since the prototypes are parametrized by the vector of design variables, it is possible to apply the spectral optimization algorithm to select a dominant prototype among the concepts. We are currently evaluating integrating the algorithm into the set of search algorithms.

The general approach to prototype selection presented in Section 1.4 and the representation of prototypes as vectors in space represents the formalization of a basic principle in information retrieval [18]; the similarity of two items of information is defined with respect to their similar features. The method explored here introduces a scheme that tries to provide a basis for determining the closedness in feature similarity, the assumption being that two prototypes are similar to the extent that they contain the same features. The spectral optimization algorithm computes the set of similarity parameters that allows the comparison of feature similarity, yielding the information to perform prototype selection. The vector space and spectral optimization idea introduced here seems to provide a good starting point for dealing with multi-objective prototype selection.

Acknowledgments

The authors wish to thank the Scientific Research Labs at Ford Motor Company for the financial support of this research. In addition, we wish to acknowledge our corporate Autodesk, Inc. for their ongoing support of our research.

Appendix

We present an abbreviated proof for the uniqueness of the optimal approximation of an arbitrary vector **V** and for the coefficients α.

Proposition : Let **V*** $\in$ **R** where **R** is a subspace which spans the space $\{b_1, b_2 \ldots b_n\}$. In this context, **R** spans the hyperspace of all possible designs. If **V** - **V*** $\perp$ **R**, then **V*** is an optimal approximation. Furthermore, the optimal approximation is unique. The approximation is given by Equation 1 and the coefficients by Equation 2.

Proof :

We shall show that there does not exist another vector **x** $\in$ **R** which as an error norm $\|\mathbf{V} - \mathbf{x}\|^2$ smaller than the error norm $\|\mathbf{V} - \mathbf{V^*}\|^2$. This implies that **V*** is the optimal approximation of **V**. The proof follows:

$$\|V - x\|^2 = \left\|V - V^* + V^* - x\right\|^2$$

$$\|V - x\|^2 = \left\|V - V^*\right\|^2 + \left\|V^* - x\right\|^2 \text{ by Pythagoras}$$

$$\|V - x\|^2 \geq \left\|V - V^*\right\|^2 \text{ by Triangle Inequality}$$

Therefore, unless **x** = **V***, there cannot exist another vector which is better approximates **V.**

Now, show that **V*** is unique. Let **x** and **V*** be equally optimal approximations. By the last equation in the proof, **x** must equal **V***. Therefore, **V*** is unique.

Finally, derive the transition matrix. We want to show that the error vector **e** $\equiv$ **V** - **V*** $\perp$ **R**. We shall show this result by proving that **e** $\perp$ b_i . If **e** $\perp$ b_i, then the inner product of <**e**, b_i> = 0. If the error vector is orthogonal to each of the b_i then it must be orthogonal to **R**. In particular, the case <e, $\mathbf{b}_i$> = 0 is proved. The extension to the other b_i is clear. Recalling the definition of b_i,

$$< e, b_1 >=< V - V^*, b_1 >=< V - b_1 > - < V^* - b_1 >$$

$$=< V - b_1 > - \sum_{i=1}^{k} \alpha_i < b_i, b_1 >$$

$$=< V - b_1 > - \left[< b_1, b_1 >, < b_1, b_2 >, \cdots < b_1, b_k >\right] \begin{bmatrix} \alpha_1 \\ \vdots \\ \alpha_k \end{bmatrix}$$

$$=< V - b_1 > - \begin{bmatrix} 1 & 0 & \cdots & 0 \end{bmatrix} \begin{bmatrix} < b_1, b_1 > & \cdots & < b_1, b_k > \\ \vdots & \ddots & \vdots \\ < b_k, b_1 > & \cdots & < b_k, b_k > \end{bmatrix} \begin{bmatrix} \alpha_1 \\ \vdots \\ \alpha_k \end{bmatrix}$$

$$=< V - b_1 > - \begin{bmatrix} 1 & 0 & \cdots & 0 \end{bmatrix} BB^{-1} \begin{bmatrix} < V - b_1 > \\ \vdots \\ < V - b_k > \end{bmatrix} = 0$$