Proceedings of the ASME 2009 International Design Engineering Technical Conferences & Computers and Information in Engineering Conference
IDETC/CIE 2009
August 30 - September 2, 2009, San Diego, California, USA

DETC2009-86462

SEPARATING DESIGN OPTIMIZATION PROBLEMS FOR BOUNDED RATIONAL DESIGNERS

Jeffrey W. Herrmann
University of Maryland
College Park, Maryland, U.S.A.

ABSTRACT

This paper presents a method for assessing the quality of progressive design processes that seek to maximize the profitability of the product that is being designed. The proposed approach uses separations, a type of problem decomposition, to model progressive design processes. The subproblems in the separations correspond roughly to phases in the progressive design processes. We simulate the choices of a bounded rational designer for each subproblem using different search algorithms. We consider different types and versions of these search processes in order to determine if the results are robust to the decision-making model. We use a simple two-variable problem to help describe the approach and then apply the approach to assess motor design processes. Methods for assessing the quality of engineering design processes can be used to guide improvements to engineering design processes and generate more valuable products.

INTRODUCTION

In general, design optimization determines values for design variables such that an objective function is optimized while performance and other constraints are satisfied [1, 2, 3]. The study of design optimization has yielded many useful techniques for solving optimization problems. Design engineers may use these techniques during a design process whenever appropriate mathematical models are available to represent the objective function and constraints in terms of the design variables [2].

Decomposition-based design optimization methods divide a large optimization problem into a set of smaller subproblems but require multiple iterations to converge to a feasible, optimal solution. Model coordination and goal coordination are two common methods for the decomposition of large scale design optimization problems [4, 5]. Multidisciplinary optimization (MDO) problems have been the focus of decomposition approaches such as the bi-level integrated system synthesis (BLISS) approach [6], analytical target cascading [7-9], collaborative optimization [10], and coupled subspace optimization (CSSO) [11, 12]. Yoshimura et al. [13] decompose a multi-objective optimization problem into a hierarchy of problems that have two objectives.

One can assess the quality of a decomposition-based design optimization method by evaluating the solution that is generated by the method using the objective function of the original design optimization problem and comparing that to the optimal value. For instance, if a decomposition-based design optimization method is used to solve an aircraft design optimization problem in which the objective function is to minimize the aircraft's gross takeoff weight, then one can assess the quality of that method by comparing the gross takeoff weight of the method's solution to the optimal gross takeoff weight (found by solving the original design optimization problem).

Decomposition-based design optimization methods are a form of problem decomposition, which is one type of decomposition in engineering design [14]. A second type is product decomposition, which divides the physical elements of a product (or system) into subassemblies and components. A third type is process decomposition, which divides the entire design task (from gathering requirements to detailed design) into the various activities that form engineering design processes.

In this paper, we will consider a certain class of engineering design processes that we call progressive design. A *progressive design process* is an engineering design process that creates a product or system design through a series of distinct phases. (Thus, this term would not cover prototype-based design processes that iterate through generate-build-test cycles.) Each phase generates intermediate results by making decisions about different aspects of the design and generates increasingly detailed information. (The name reflects the similarity to a progressive die, which makes an increasingly complex part through a series of punches.) Pahl and Beitz [15],

 Copyright © 2009 by ASME

Asimow [16], Ullman [17], and Ulrich and Eppinger [18] are among those presenting progressive design processes, and such processes are commonly found in industry [19, 20].

Previous work on decision-making in engineering design has shown that design occurs via a series of decisions and that engineering design organizations are decision-making systems [21, 22]. Some decisions may be done sequentially while others occur concurrently, and different types of decision-making processes (such as designers who cooperate fully, designers who make isolated decisions, and sequential decision-making processes) can occur [23]. Research on this topic has focused primarily on the decision-making that sets parametric design variables within one phase of a system design project (e.g., the conceptual design of an aircraft or sizing a pressure vessel or electric motor).

Researchers have developed models of design processes that focus on the activities that need to be done, as in Gantt charts, the PERT and critical path methods, IDEF, the design structure matrix, Petri nets, and signposting [24]. Such models have been used to estimate the cost and duration of design processes [25-29].

Some research has focused instead on assessing the quality of the design processes by evaluating the quality of the product designs that are generated, just as those studying decomposition-based design optimization methods assess the quality of those methods by evaluating the solutions that the methods find.

Chang and Ward [30] propose a method for simultaneous decision-making of an engineering design team that focuses attention on the cost of waiting for information to make better decisions and the cost of making a wrong decision based on incomplete information. The simultaneous decision-making considered in their approach is not necessarily a good model of a progressive design process, however.

Krishnan et al. [31] consider the quality loss of engineering design processes, which they model as sequences of decisions made by different members of the engineering design team (see also [32]). The quality loss measures the distance of the resulting design from the optimal design (determined by making all of the decisions at the same time). They calculate the quality loss for different sequences of decisions in a DC motor design process. The quality loss of a sequence is determined by comparing each member's decision, which is constrained by the decisions make earlier in the sequence, to that member's optimal. Each member makes his decision once and has a different "optimal design" because they have different objectives. There is no bottom-line objective that all of the members seek to optimize.

Lewis and Mistree [23] consider the design of a pressure vessel and the design of a passenger aircraft. For both problems, they compared the quality of the solutions from different decision-making processes, which were modeled as two-player games. The objectives of the pressure vessel designers are to minimize weight and to maximize volume; the objectives of the passenger aircraft designers are to minimize deviations from aerodynamic and weight goals. In neither case is a more fundamental objective like profitability considered.

An important development in recent years is the emergence of design for market systems approaches [33] and other work motivated by the decision-based design (DBD) framework [34], which includes enterprise models that add variables from the marketing and manufacturing domains to models with conceptual design variables and adapt existing decomposition techniques to solve them [35-37]. This perspective suggests that the profitability of a product (a measure of its performance in the marketplace) should be used to choose between different design options. This builds on the earlier suggestions of writers such as Smith and Reinertsen [38], who developed product profit models and used them to derive tradeoff rules to guide design decision-making. An important implication of this perspective is that it provides a single enterprise-level objective for evaluating the quality of product designs, which, as discussed above, is a way to assess the quality of an engineering design process.

If organizations had a method for assessing the quality of engineering design processes, they could use this to guide improvements to their engineering design processes and generate more valuable products. In addition, one could assess the quality of design decision-making tools by considering how well they improve the entire engineering design process, not just one step. The purpose of the study described here is to develop such a method.

In order to do this, we assume that the objective of an engineering design process is to maximize the profitability of a new product. We then model different progressive design processes using separations, a special class of problem decompositions that are significantly different from typical decomposition-based design optimization methods [39].

Because humans are bounded rational decision-makers, we do not use optimization to solve the subproblems in the separation (as done in [39]). Instead, like Gurnani and Lewis [40], we model the designer's bounded rational design choices as a random process. In [40], the values chosen by the designer were determined by a random error term. In this work, however, the values are chosen by random search processes. We consider different types and versions of these search processes in order to determine if the results are robust to our decision-making model. We use a simple two-variable problem to help describe the approach and apply the approach to assess motor design processes.

MODELING DESIGN PROCESSES

In this study, we use separations to model progressive design processes. A *separation* is a type of problem decomposition that divides a design optimization problem into a set of subproblems, solves each subproblem once, and produces a feasible solution without iterative cycles [39]. The ideal separation produces an optimal solution to the original problem. Unfortunately, not all separations do. Both separations and decomposition-based design optimization

Copyright © 2009 by ASME

methods replace a large design optimization problem with a set of subproblems. Separations are distinct because they do not solve a second-level problem to coordinate the subproblem solutions in an iterative manner.

In their work on sequential decision-making, Krishnan et al. [31] use the term *decision order* to describe a decision process in which each member of a cross-functional team makes his decision exactly once, in a predetermined. Such a sequence is one type of separation.

A leader-follower (or Stackelberg) game is also a simple separation. In this scenario, one player makes his decision, and then the other player makes his decision. Lewis and Mistree [23] studied this as one model of collaborative decision making.

Previous work evaluated the quality of different separations by using optimization to solve each subproblem [39]. If the subproblems are correctly formulated, the separation yields an optimal solution. The quality of approximate separations depends upon the constraints and objectives used in the subproblems.

In this paper, we follow up on the idea suggested in [39] that separation could be used to model an engineering design process. We will study separations corresponding to different types of progressive design processes. If decision-makers always optimized, then separation should, except under certain conditions, lead to a quality loss [31, 39]. Because real-world decision makers do not optimize, we will evaluate these separations by using a model of a bounded rational designer to find solutions to each subproblem.

MODELING BOUNDED RATIONALITY

Most studies of engineering decision-making processes assume that each engineer optimizes his objective function. For example, to design a pressure vessel, Lewis and Mistree [23] formulated the problem as a two player game in which the first player, who wishes to maximize the volume, controls the diameter and length and the second player, who wishes to minimize the weight, controls the thickness. Chen and Li [41] modeled the interactions of teams in concurrent parametric design under the assumption that each team optimized their aspect of the complete design. In the decision orders of Krishnan et al. [31], each team member selects values for their design variables in order to optimize their individual objective function.

It is well-known that real-world decision-makers do not optimize because of limits on their problem-solving capacity. This concept is known as bounded rationality. Bounded rationality reflects the observation that, in most real-world cases, decision-makers have limited information and limited computational capabilities for finding and evaluating alternatives and choosing among them [42-44]. A decision-maker cannot perfectly evaluate the consequences of the available choices. This prevents complete and perfect optimization. The study of procedures besides optimization to choose between alternatives has been defended by appealing to bounded rationality, as in the case of pairwise comparisons of a finite set of alternatives [45].

Satisficing and fast and frugal heuristics are two models of bounded rationality [43]. Bounded rational decision-makers may search until they find something that meets their requirements (satisficing), or they may use fast and frugal heuristics that search a limited set of objects and information and make choices using rules that are easy to compute (and therefore quick). We consider this second type of bounded rationality in this work but will study satisficing in future studies.

Gurnani and Lewis [40] studied collaborative, decentralized design processes in which the models of the individual decision-makers (the designers) were based on the ideas of bounded rationality. In their model, the value chosen by each designer was determined by randomly sampling from a distribution around the (locally) "optimal" solution. This model is meant to represent the mistakes that designers make due to bounded rationality. Their results show that incorporating bounded rationality leads to more desirable solutions in a collaborative, decentralized design process in which the designers had different objectives and no way to coordinate their activities.

In this paper we will take a different approach to modeling bounded rationality. To begin, we will first consider models of organizational decision-making and problem solving and then consider cognitive models of human problem solving.

In each phase of a progressive design process, some aspect of the design needs to be determined. That is, the designer must make a decision. In practice, the designer does not have complete information about the available alternatives and their performance on the relevant attributes. Thus, the decision-maker must employ some process for generating and evaluating alternatives. (Note that we are studying situations that involve searching for alternatives and the information needed to evaluate them, not simple choices or perfect information games and puzzles like chess and Sudoku.)

The *incremental decision process model* describes the activities that occur during decision-making processes in product development, facility design, procurement, and policy setting [46]. This decision-making process (also known as the strategic decision-making process) is not a simple sequence of tasks but involves iterating between the following types of activities: recognition, diagnosis, search, screen, design, judgment, analysis, bargaining, and authorization.

March and Simon [44] describe the general characteristics of human problem-solving in organizational decision-making. The first characteristic is that making a complex decision involves making a large number of small decisions. The second characteristic is that problem-solving has a hierarchical structure in which solving any problem goes through phases that, in turn, require solving more detailed subproblems. The general concept of separation is related to these two characteristics. The third characteristic is that problem-solving consists of searching for possible solutions [47]. The fourth

Copyright © 2009 by ASME

characteristic is that problem-solving includes screening processes that evaluate the solutions that are found. The fifth characteristic is that problem-solving has not only random components (such as finding and evaluating solutions) but also a procedural structure that allows it to yield good solutions. The proposed model of a bounded rational designer is motivated by these last three characteristics.

Wang and Chiew [48] describe human problem solving as a higher-layer cognitive process that can be considered as a search process, though it requires other cognitive processes such as abstraction, analysis, synthesis, and decision-making. Their model of a generic problem solving process is a search that iteratively generates and evaluates potential solutions.

Thus, we see that decision-making in the context of product development is equivalent to problem solving by organizations and individual humans and, furthermore, that such problem solving is essentially a search.

An important aspect of bounded rationality is that the resources and time available for problem-solving are limited. Consequently, our model of a bounded rational designer incorporates limits that will constrain the amount of time available for the search and the accuracy of the evaluation of a solution.

Therefore, we will use a class of search algorithms that identify and evaluate solutions to model the choices of a bounded rational designer. Each search has random components (either randomly selecting a solution or randomly moving to a point near the existing solution), and the evaluation of a solution has a random error that represents inaccuracies due to limited time and knowledge. The procedural structure of the model attempts to compensate for this randomness by keeping track of the "best" solution found so far. (The "best" requires quotes because the evaluation of the solution may be inaccurate.) Finally, the search is limited to a fixed number of solutions.

This study considers two types of searches: random sampling and local search. Within each type, the search is characterized by two parameters: *N*, the search effort, and *E*, the maximum relative error. Let *X* be the vector of design variables with lower and upper bounds *L* and *U*. In addition, the search may need to satisfy one or more constraints *C* in addition to the lower and upper bounds. Let *f*(*X*) be the objective function.

The random sampling search works as follows:

Do the following step *N* times: Randomly select *X* between *L* and *U*. If *X* is not feasible with respect to *C*, then "repair" *X* by projecting it into the feasible space and replacing *X* with the repaired solution. Select the relative error ε from a uniformly distributed random variable with the range [-*E*, *E*]. Set *g*(*X*) = (1 + ε)*f*(*X*). If *g*(*X*) is the best function evaluation found so far, keep *X* as the best solution found so far.

The local search works as follows:

Randomly select *X* between *L* and *U*. If *X* is not feasible with respect to *C*, then "repair" *X* by projecting it into the feasible space and replacing *X* with the repaired solution. Select the relative error ε from a uniformly distributed random variable with the range [-*E*, *E*]. Set *g*(*X*) = (1 + ε)*f*(*X*). Do the following step *N* times: Let *D* = (1/20)(*U* – *L*). Randomly select a step *S* between -*D* and *D* and let the new solution *Y* = *X* + *S*. If *Y* is not feasible with respect to *C*, then "repair" *Y* by projecting it into the feasible space and replacing *Y* with the repaired solution. Select the relative error ε from a uniformly distributed random variable with the range [-*E*, *E*]. Set *g*(*Y*) = (1 + ε)*f*(*Y*). If *g*(*Y*) is better than *g*(*X*), then set *X* = *Y*.

Both searches return *X*, the "best" solution found so far. After the search ends, we will, for evaluation purposes, calculate *f*(*X*), the true value of the objective function at this point.

It is important to note that these searches are meant to represent a bounded rational designer. These algorithms should not be compared to state-of-the-art techniques for solving design optimization problems. Like [40], we are modeling the designer's bounded rational design choices as a random process. In [40], the values chosen by the designer were determined by a random error term. In our approach, on the other hand, the values are chosen by a random search process. To model a bounded rational designer who is using a type of fast and frugal heuristic, these searches have simple rules to stop the search (when the number of solutions evaluated equals *N*) and to choose a solution (whether it is better than the best found so far).

For instance, consider a simple two-variable problem. There are two design variables, x_1 and x_2, with the bounds $1 \le x_1 \le 10$ and $1 \le x_2 \le 10$ The objective function, which we wish to minimize, is $f\left(x_1, x_2\right) = x_2 / x_1 + \frac{1}{20}\left(x_2 - 5\right)^2$. The optimal value occurs at (x_1, x_2) = (10, 4), with *f*(10, 4) = 0.45. This point is marked by a grey square with a black X in Figure 1.

First, we consider a search for points (x_1, x_2). For the purposes of this example, we generated 20 points by sampling from uniform distributions for both x_1 and x_2 (both on the interval [1, 10]). This yielded the points in Table 1, and they are also marked by diamonds in Figure 1. The values of $f\left(x_1, x_2\right)$ are given as well, and the best point is (8.8046, 5.2671), with *f*(8.8046, 5.2671) = 0.6018. This point is marked with a square in Figure 1.

Next we consider a process that employs two concurrent searches: one for x_1 and one for x_2. The search for x_1 seeks to minimize $1/x_1$, and the search for x_2 seeks to minimize $\left(x_2 - 5\right)^2$. For each variable, we used the same 20 samples that were generated before. The best x_1 = 9.3261, and the best x_2 = 5.2671. This point is marked with a grey triangle in Figure 1, and *f*(9.3261, 5.2671) = 0.5192.

Copyright © 2009 by ASME

TABLE 1. SAMPLE POINTS FOR EXAMPLE.

Point	x_1	x_2	$f(x_1, x_2)$
1	6.9567	9.4910	2.3728
2	6.7216	4.7429	0.7089
3	7.7413	6.8977	1.0711
4	2.0509	7.5269	3.9894
5	7.0820	6.7763	1.1146
6	2.8837	5.7263	2.0121
7	3.9476	5.8739	1.5262
8	2.2510	2.5185	1.4267
9	3.6916	7.5195	2.3543
10	8.8191	2.2350	0.6357
11	5.4328	9.2444	2.6023
12	4.8252	6.7868	1.5662
13	2.9704	4.2774	1.4661
14	6.1129	7.7658	1.6529
15	5.0185	6.8124	1.5217
16	3.1707	4.8280	1.5242
17	8.8046	5.2671	0.6018
18	9.3261	1.7465	0.7165
19	8.8042	1.1194	0.8801
20	3.6420	1.4091	1.0316

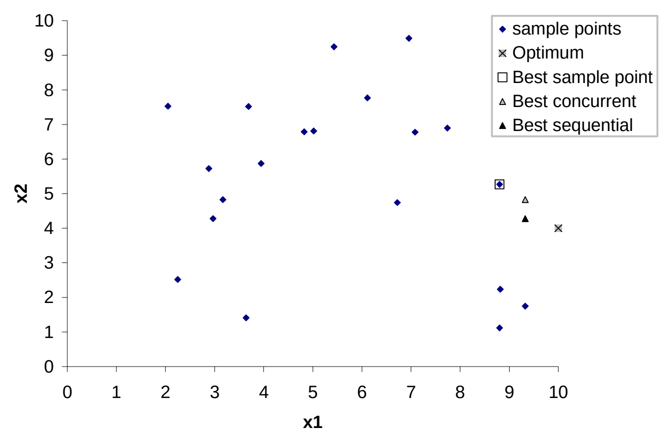

FIGURE 1. SAMPLE POINTS AND SOLUTIONS FOR TWO-VARIABLE EXAMPLE.

Finally, we consider a process that is a sequence of two searches: the first for x_1 and the second for x_2. The search for x_1 seeks to minimize $1/x_1$, and the search for x_2 seeks to minimize $f(x_1, x_2)$. For each variable, we used the same 20 samples that were generated before. The best x_1 = 9.3261. Given this value for x_1, the best x_2 = 4.2774. This point is marked with a black triangle in Figure 1, and f(9.3261, 4.2774) = 0.4848.

Note that, for a bounded rational decision-maker, the decision-making processes (separations) that divided the original optimization problem into two subproblems found better solutions than the process that considered both variables at the same time. The solution of the concurrent decision-making process was not as good because the search for x_2 considered only half of the objective function (because there was no value of x_1 present). The solution of the sequential decision-making process was closer to the optimal solution.

EXAMPLE: MOTOR DESIGN OPTIMIZATION

A universal electric motor example originally developed by Simpson [49] will be used to demonstrate the separations and our model of a bounded rational designer. Simpson used this example to demonstrate new techniques in product family design. The following example (which is the same as the one in [39]) ignores the product family aspect and deals with only a single motor design that should meet given power and torque requirements.

The optimization model for the universal electric motor problem includes nine design variables, four customer attributes, twenty-three intermediate engineering attribute calculations, six constraints, and seven fixed engineering parameters. The derivations of the equations and other background information on universal electric motors can be found in [49]. The nomenclature and equations for the design variables, fixed model parameters, customer attributes, and constraints are listed below. Annex A lists the intermediate engineering attributes. Table 2 lists the product design variables and their bounds. The price p also must be determined.

The four customer attributes are the torque T (in Nm), the power P (in watts), the efficiency η, and the mass M (in kg). They are calculated from the design variables and the engineering attributes as follows:

$$\begin{aligned} T &= K\varphi I \\ P &= P_{\text{in}} - P_{\text{out}} \\ \eta &= P / P_{\text{in}} \\ M &= M_w + M_s + M_a \end{aligned} \qquad (1)$$

The product's power should equal 300 watts, and its torque should equal 0.05 Nm.

To predict demand, we used a logit model with independent and identically distributed error terms. (For more about this type of model in particular and discrete choice analysis in general, see [50].) In this aggregate model, demand depends upon the additive utility function ν, which is calculated from the utility functions for the mass, efficiency, and price attributes [51]. The total demand (d) is the population size (s) multiplied by the probability that a consumer will select this design. We set s = 1,000,000.

$$\begin{aligned} \Psi_1(M) &= 0.5(1 - M) \\ \Psi_2(\eta) &= \eta - 0.5 \\ \Psi_3(p) &= \frac{25 - 4p}{15} \\ v &= \Psi_1(M) + \Psi_2(\eta) + \Psi_3(p) \\ d &= se^v / \left(1 + e^v\right) \end{aligned}$$

Copyright © 2009 by ASME

TABLE 2. BOUNDS ON PRODUCT DESIGN VARIABLES.

Variable	Definition	Bounds	units
N_c	Turns of wire (armature)	[100, 1500]	turns
N_s	Turns of wire (stator), per pole	[1, 500]	turns
A_{aw}	Cross sectional area of armature wire	[0.01, 1.0]	mm^2
A_{sw}	Cross sectional area of stator wire	[0.01, 1.0]	mm^2
r_o	Outer radius (stator)	[0.01, 0.1]	m
t_s	Thickness (stator)	[0.0005, 0.01]	m
I	Electric current	[0.1, 6]	A
L	Stack length	[0.01, 0.2]	m

To model the product cost, we modified the cost equations that were originally derived in Wassenaar and Chen [52]. The design cost C_D is assumed to be fixed at $500,000 while the total material cost C_M, labor cost C_L, and capacity cost C_K vary with demand and engineering attributes.

$$C_D = 500,000$$

$$C_M = d\left(M_w C_{copper} + \left(M_s + M_r\right)C_{steel}\right)$$

$$C_L = \frac{3}{7}C_M$$

$$C_K = 50\left(\left(d - 500,000\right)/1000\right)^2$$

The profit Π of a motor design is a function of the demand (d), price (p), and the costs discussed above.

$$\Pi = dp - \left(C_D + C_M + C_L + C_K\right)$$

The motor design problem has eight design variables, but we wish to find motor designs that satisfy given constraints on the power and torque. Given values for the first six design variables, it is possible to find values for I and L so that product satisfies the power and torque requirements. (Annex B derives the required equations.) Thus, we need to find values for only six of the eight design variables (and the price). These we call the independent design variables.

We will consider five different separations (formulations) of the motor design optimization problem as models for three different progressive design processes (see also Table 3). The first progressive design process has only one phase that determines the design and the price simultaneously in order to maximize profitability (this corresponds to the enterprise models and design for market systems approaches inspired by the DBD framework). The second is a simplified version of a traditional design process with two phases: the first phase determines targets for the key product attributes, and the second phase attempts to design a product to meet those attributes. The price is set to maximize the profitability assuming that the attributes will be met. The third is a performance-centered process that has two phases: the first phase designs the product in order to optimize its performance (or cost), and the second phase sets an appropriate price.

TABLE 3. DESIGN PROCESSES AND SEPARATIONS.

Progressive design process	Separation(s)
One phase: determine design and price simultaneously	A
Two phases: (1) determine targets, (2) attempt to meet the targets.	B
Two phases: (1) optimize performance, (2) set price to maximize profitability.	C (minimize mass) D (maximize efficiency) E (minimize material cost)

The first separation (A) is an all-at-once formulation in which the designer must determine values for the six independent design variables and the price simultaneously in order to maximize profitability.

In the second separation (B), the designer solves three subproblems. First, the designer determines target values for mass and efficiency in order to maximize the utility $\Psi_1(M) + \Psi_2(\eta)$. The lower and upper bounds on mass are 0 and 4 kg, and the lower and upper bounds on efficiency are 0.30 and 0.97. These bounds were chosen as they span the range of feasible, desirable values for these attributes. We include the following constraint in order to restrict the search to reasonable combinations of mass and efficiency: $M \geq 0.02/\left(1-\eta\right)$. If the bounded rational designer selects a solution does not satisfy this constraint (the mass is too small), the value of M is increased to $0.02/\left(1-\eta\right)$ so that the solution becomes feasible. (The policy to “repair” an infeasible solution, instead of discarding it, is part of our model of the bounded rational designer.)

In the second subproblem in separation B, given the target values for mass and efficiency, the designer sets the price in order to maximize (estimated) profitability. Estimating the profitability requires estimating the total material cost as dMC^*, where C^* = $2 per kg is an average material cost.

In the third subproblem in separation B, the designer determine values for the six independent design variables in order to minimize the deviation from the mass and efficiency targets M_T and η_T. The objective function of the third problem is the following value:

$$\left(1 - \frac{M}{M_T}\right)^2 + \left(1 - \frac{\eta}{\eta_T}\right)^2$$

The third, fourth, and fifth separations each have two subproblems. In separation C, the designer first determines values for the six independent design variables in order to minimize the mass of the motor. Then, the designer sets the price in order to maximize profitability.

In separation D, the designer first determines values for the six independent design variables in order to maximize the

Copyright © 2009 by ASME

efficiency of the motor. Then, the designer sets the price in order to maximize profitability.

In separation E, the designer first determines values for the six independent design variables in order to minimize the material cost per motor, determined as $M_w C_{copper} + \left(M_s + M_r\right) C_{steel}$. Then, the designer sets the price in order to maximize profitability.

To evaluate these separations, we model the designer using both types of search (sampling and local search), two levels of search effort (500 and 1000 solutions per search), and three values for maximum error (0, 1%, and 10%). We assume that the maximum error and search effort are the same across all of the subproblems in a given separation. The levels of search effort were chosen after preliminary results showed that the profitability of the solutions continued to increase significantly until about 500 samples; after that the profitability increased more slowly. We will consider other stopping rules in future work.

For each combination of separation, search type, search effort, and maximum error, we ran 25 replications. The results are described in Section 5. We also used optimization to search for the most profitable solution, which will provide a benchmark for the results. The best solution found had a profit of $3,998,500.

RESULTS

The results show that some progressive design processes yield much better solutions than others, as shown in Tables 4 to 9. The quality of the solutions found depends upon the type search and the inaccuracy. Inaccurate evaluations in a local search may lead it in the wrong direction or prevent it from leaving poor solutions. The very large confidence intervals reflect the fact that roughly half of the replications yielded extremely unprofitable solutions. The other replications yielded high-quality solutions.

TABLE 4. RESULTS OF LOCAL SEARCH FOR DIFFERENT VALUES OF SEARCH EFFORT. MAXIMUM ERROR = 0. 90% CONFIDENCE INTERVALS ON THE AVERAGE PROFIT.

Separation	500	1000
A	3,508,302 ± 911,954	3,742,671 ± 338,653
B	3,840,919 ± 32,529	3,845,258 ± 23,194
C	3,156,807 ± 402,115	3,105,345 ± 437,144
D	3,016,544 ± 132,751	3,025,348 ± 99,276
E	3,107,823 ± 458,650	3,016,948 ± 483,558

TABLE 5. RESULTS OF LOCAL SEARCH FOR DIFFERENT VALUES OF SEARCH EFFORT. MAXIMUM ERROR = 1%. 90% CONFIDENCE INTERVALS ON THE AVERAGE PROFIT.

Separation	500	1000
A	-3,713,717 ± 14,378,068	-3,602,121 ± 14,547,629
B	-4,253,888 ± 14,672,901	-3,567,488 ± 14,600,395
C	-4,632,921 ± 14,042,293	-4,656,971 ± 14,001,649
D	-4,798,838 ± 13,841,660	-4,034,374 ± 13,882,237
E	-4,628,350 ± 14,048,658	-4,690,512 ± 13,946,395

TABLE 6. RESULTS OF LOCAL SEARCH FOR DIFFERENT VALUES OF SEARCH EFFORT. MAXIMUM ERROR = 10%. 90% CONFIDENCE INTERVALS ON THE AVERAGE PROFIT.

Separation	500	1000
A	-7,809,342 ± 13,132,897	-5,709,928 ± 14,318,599
B	-6,938,373 ± 14,101,412	-5,582,774 ± 14,598,025
C	-7,834,003 ± 13,146,632	-7,213,469 ± 13,458,660
D	-8,333,808 ± 12,151,554	-6,888,305 ± 13,163,572
E	-7,882,243 ± 13,023,629	-7,249,029 ± 13,369,865

TABLE 7. RESULTS OF SAMPLING FOR DIFFERENT VALUES OF SEARCH EFFORT. MAXIMUM ERROR = 0. 90% CONFIDENCE INTERVALS ON THE AVERAGE PROFIT.

Separation	500	1000
A	3,371,776 ± 466,800	3,506,449 ± 384,085
B	3,720,915 ± 160,931	3,756,515 ± 122,944
C	3,416,876 ± 597,613	3,495,813 ± 493,674
D	3,026,388 ± 296,620	2,879,891 ± 769,956
E	3,420,159 ± 603,670	3,505,570 ± 508,263

TABLE 8. RESULTS OF SAMPLING FOR DIFFERENT VALUES OF SEARCH EFFORT. MAXIMUM ERROR = 1%. 90% CONFIDENCE INTERVALS ON THE AVERAGE PROFIT.

Separation	500	1000
A	3,371,776 ± 466,800	3,506,449 ± 384,085
B	3,717,794 ± 164,009	3,743,194 ± 133,607
C	3,413,388 ± 598,051	3,489,527 ± 488,074
D	2,788,795 ± 1,740,672	2,995,104 ± 452,042
E	3,416,574 ± 604,454	3,499,289 ± 503,219

TABLE 9. RESULTS OF SAMPLING FOR DIFFERENT VALUES OF SEARCH EFFORT. MAXIMUM ERROR = 10%. 90% CONFIDENCE INTERVALS ON THE AVERAGE PROFIT.

Separation	500	1000
A	3,349,374 ± 450,858	3,453,575 ± 432,616
B	3,620,275 ± 253,152	3,667,113 ± 239,461
C	3,394,254 ± 597,856	3,507,559 ± 397,218
D	2,515,286 ± 1,873,824	2,584,092 ± 1,872,540
E	3,427,512 ± 511,251	3,495,834 ± 514,534

When the evaluations are perfectly accurate, the search effort during each local search made a significant difference only in separations A, C, and E. The worst results were found using separation D, which maximizes efficiency. The results with local search in separation B are significantly better than those with local search in any other separation. When the search effort equals 1000 and the maximum error is 0, the 95% confidence interval on the difference between separations B and A is [19,019, 186,155].

The results from separations with random sampling show a different pattern with regard to accuracy level (see Tables 7 to 9). In most cases, increasing inaccuracy (changing the maximum error from 0 to 1% to 10%) made only small changes to the quality of the solution generated. The largest changes occurred when using separation D, but this separation generally gave poor solutions in all cases. The changes, though small, were sometimes statistically significant. For instance, for separation B, the changes from 0 to 1% were not statistically

Copyright © 2009 by ASME

significant, but the changes from 1% to 10% were. Overall, the results with random sampling in separation B are significantly better than those with random sampling in any other separation. The results from separations C and E were close to those from separation A, and the results from separation D were much worse. When the search effort equals 1000 and the maximum error is 0, the 95% confidence interval on the difference between separations B and A is [155,878, 344,256], and the 95% confidence interval on the difference between separations B and D is [726,639, 1,026,610].

When random sampling is used, the impact of increasing the search effort was largest in separation A. This was the only separation in which the difference was statistically significant as well. In the other separations, increasing the search effort from 500 to 1000 did not make a statistically significant difference in the results of the random sampling. This may be because the subproblems in the other separations have fewer variables and a smaller search space than the one in separation A (which has six independent design variables and the price).

If there are inaccuracies in evaluating the function, then random sampling finds better solutions than local search. If there are no inaccuracies, then, in separation B, which is the best separation under both types of search, local search finds more profitable solutions that are near optimal. When the search effort equals 1000 and the maximum error is 0, the 95% confidence interval on the difference between local search and sampling for separation B is [56,620, 120,866].

Two other general observations: the results of separation E were quite similar to those of separation C, because the mass and the material cost per motor are highly correlated. Finally, separation D, as noted before, yielded quite poor results. Maximizing efficiency is not a good objective.

SUMMARY AND CONCLUSIONS

This paper presented a method for assessing the quality of progressive design processes. The proposed approach uses separations, a type of problem decomposition, to model progressive design processes. The subproblems in the separations correspond roughly to phases in the progressive design processes. We modeled the choices of a bounded rational designer for each subproblem using different search algorithms.

We considered two types of searches: a sampling search that randomly selects feasible solutions and keeps the best found so far and a local search that randomly selects a solution near the current solution and moves to the new solution if it is better than the current solution.

The results show that, for our motor design example, a design process that sets the attribute values first and then searches for a design that meets those targets is a superior approach and generates better solutions than a process that attempts to solve the profit maximization problem all-at-once. Other design processes perform poorly. The relative performance is robust and holds across a range of different variations of the model of the bounded rational designer. In addition, the results show that the quality of different types of searches depends upon the presence of error in the evaluation of a solution.

Note that these results are not meant to imply that designers should use any of these algorithms explicitly to solve design optimization problems. They are models of how a bounded rational designer makes choices, and their purpose is to help assess the quality of different design processes.

We began by considering how an organization could assess the quality of engineering design processes and discussed how the emergence of the design for market systems approach suggests that profitability is an appropriate objective function for evaluating design process quality. The approach presented here uses this objective, unlike previous approaches that focused on less-fundamental objectives. Moreover, instead of assuming that designers optimize, we take a more realistic position that designers are bounded rational decision-makers. We use a random process to model their design choices as part of the assessment procedure. The results show that some (but not all) progressive design processes lead to designs that are more profitable than those generated by attempts to maximize profitability by considering all of the design variables simultaneously.

These results therefore provide some insight into the benefits of progressive design processes. Namely, because human decision-makers cannot optimize, well-designed progressive design processes are the best way to generate profitable product designs. Moreover, the objectives and constraints of the decision-making activities in a progressive design process are influential factors. Therefore, modifying these can significantly improve or degrade the quality of a progressive design process.

This research, unlike previous work, is ultimately concerned with the overall engineering design process, not with decision-making within a phase of the process. Although progressive design processes for complex products are extremely complicated, it is hoped that models like the ones presented here could be used to help organizations improve their design processes. Models that include the key decision-making activities and most important design variables may provide useful insights.

The separations studied in [39] were used to solve design optimization problems, and optimization was used to solve each subproblem. Separations reduced the computational effort, and some separations generated solutions that were nearly as profitable as those found by solving the all-at-once formulation.

In this paper, separations are used as models of progressive design processes, and random searches (rather than optimization) are used to find solutions to the subproblems. The results in this paper show that a bounded rational designer will generate more profitable designs by using a sequential decision-making process.

This paper builds on two important branches of the study of design decision-making. The first branch has developed the

 Copyright © 2009 by ASME

methods for profit maximization in design, and the second branch has studied the process of how decisions are made. The approach presented here is an attempt to connect these two different branches.

This paper contributes to our knowledge about design decision-making by proposing specific random search algorithms as models for the choices of a bounded rational designer (unlike [40], who used a random error term). This idea agrees with (and comes from) the importance of search in design (cf. [53]). More work is needed to consider other models based on satisficing or other types of fast and frugal heuristics.

Previous work on assessing the quality of design processes did not have an overarching objective for discussing the quality of different decision-making processes. The emerging area of design for market systems (building on the DBD framework) provides such an objective (profitability) and motivates recovering and reassessing these ideas within this new context.

A second contribution of this paper is to show that, when a bounded rational designer seeks profitable designs, separating the profit maximization problem into a set of subproblems yields a better solution than attempting to solve the profit maximization problem directly. Thus, using an appropriate progressive design process is a reasonable design for market systems strategy.

ACKNOWLEDGMENTS

Peyman Karimian wrote some of the computer code used in this work. The author also appreciates the useful comments of anonymous reviewers.

REFERENCES

[1] Arora, J.S., 2004, Introduction to Optimum Design, 2nd edition, Elsevier Academic Press, Amsterdam.

[2] Papalambros, P.Y., and Wilde, D.J., 2000, Principles of Optimal Design, 2nd edition, Cambridge University Press, Cambridge.

[3] Ravindran, A., Ragsdell, K.M., and Reklaitis, G.V., 2006, Engineering Optimization: Methods and Applications, 2nd edition, John Wiley & Sons, Hoboken, New Jersey.

[4] Schoeffler, J.D., 1971, "Static Multilevel Systems," Optimization Methods for Large-Scale Systems, D.A. Wismer, ed., McGraw-Hill Book Company, New York, pp. 1-47.

[5] Kirsch, U., 1981, Optimum Structural Design, McGraw-Hill Book Company, New York.

[6] Kim, H., Ragon, S., Soremekun, G., and Malone, B., 2004, "Flexible Approximation Model for Bi-Level Integrated System Synthesis," AIAA 2004-4545, 10th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference, Albany, New York, August 30 - September 1, 2004.

[7] Kim, H.M., Michelena, N.F., Papalambros, P.T., and Jiang, T., 2000, "Target Cascading in Optimal System Design," DETC2000/DAC-14265, Proceedings of DETC 2000, 26th Design Automation Conference, Baltimore, Maryland, September 10-13, 2000.

[8] Kim, H.M., Rideout, D.G., Papalambros, P.Y., and Stein, J.L., 2003, "Analytical Target Cascading in Automotive Vehicle Design," ASME Journal of Mechanical Design, **125**(3), pp. 481-489.

[9] Tosserams, S., Kokkolaras, M., Etman, L.F.P., and Rooda, J.E., 2008, "Extension of Analytical Target Cascading using Augmented Lagrangian Coordination for Multidisciplinary Design Optimization," 12th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference, 10-12 September 2008, Victoria, British Columbia, Canada.

[10] Renaud, J.E., and Gu, X., 2006, "Decision-Based Collaborative Optimization of Multidisciplinary Systems," Decision Making in Engineering Design, K. Lewis, W. Chen, and L.C. Schmidt, eds., ASME Press, New York, pp. 173-186.

[11] Bloebaum, C.L., Hajela, P., and Sobieszczanski-Sobieski, J., 1992, "Non-Hierarchic System Decomposition in Structural Optimization," Engineering Optimization, **19**(1), pp. 171-186.

[12] Wujek, B.A., Renaud, J.E., Batill, S.M., and Brockman, J.B., 1996, "Concurrent Subspace Optimization Using Design Variable Sharing in a Distributed Computing Environment," Concurrent Engineering: Research and Applications, **4**(4), pp. 361-377.

[13] Yoshimura, M., Taniguchi, M., Izui, K., and Nishiwaki, S., 2006, "Hierarchical Arrangement of Characteristics in Product Design Optimization," ASME Journal of Mechanical Design, **128**(4), pp. 701-709.

[14] Kusiak, A., 1995, "Decomposition and Representation Methods in Mechanical Design," Journal of Mechanical Design, **117**, pp. 17-24.

[15] Pahl, G., and Beitz, W., 1996, Engineering Design: a Systematic Approach, K. Wallace, ed., Springer, London.

[16] Asimow, M., 1962, Introduction to Design, Prentice-Hall, Englewood Cliffs, New Jersey.

[17] Ullman, D.G., 2003, The Mechanical Design Process, third edition, McGraw-Hill, Boston.

[18] Ulrich, K.T., and Eppinger, S.D., 2004, Product Design and Development, 3rd edition, McGraw-Hill/Irwin, New York.

[19] Cooper, R.G., 1994, "Third-Generation New Product Processes," Journal of Product Innovation Management, **11**(1), pp. 3–14.

[20] Cooper, R.G., 2008, "The Stage-Gate Idea-to-Launch Process—Update, What's New, and NexGen Systems," Journal of Product Innovation Management, **25**(3), pp. 213-232.

[21] Herrmann, Jeffrey W., and Linda C. Schmidt, 2002, "Viewing Product Development as a Decision Production System," DETC2002/DTM-34030, ASME 2002 Design Engineering Technical Conferences and Computers and

Copyright © 2009 by ASME

Information in Engineering Conference, Montreal, Canada, September 29 - October 2, 2002.

[22] Herrmann, Jeffrey W., and Linda C. Schmidt, 2006, “Product Development and Decision Production Systems,” in Decision Making in Engineering Design, K. Lewis, W. Chen, and L.C. Schmidt, editors, ASME Press, New York.

[23] Lewis, K., and Mistree, F., 1998, “Collaborative, Sequential, and Isolated Decisions in Design,” Journal of Mechanical Design, **120**(4), pp. 643-652.

[24] O’Donovan, B., Eckert, C., Clarkson, J., and Browning, T.R., 2005, “Design Planning and Modelling,” Design Process Improvement: a Review of Current Practice, J. Clarkson and C. Eckert, eds., Springer, London, pp. 60-87.

[25] Adler, P.S., Mandelbaum, A., Nguyen, V., and Schwerer, E., 1995, “From Project to Process Management: an Empirically-Based Framework for Analyzing Product Development Time,” Management Science, **41**(3), pp. 458-484.

[26] Johnson, E.W., and Brockman, J.B., 1998, “Measurement and Analysis of Sequential Design Processes,” ACM Transactions on Design Automation of Electronic Systems, **3**(1), pp. 1–20.

[27] Browning, T.R., 2001, “Applying the Design Structure Matrix to System Decomposition and Integration Problems: a Review and New Directions,” IEEE Transactions on Engineering Management, **48**(3), pp. 292-306.

[28] O’Donovan, B., Eckert, C., and Clarkson, P.J., 2004, “Simulating Design Processes to Assist Design Process Planning,” DETC2004-57612, Proceedings of the ASME 2004 Design Engineering Technical Conferences and Computers and Information in Engineering Conference, Salt Lake City, Utah, September 28-October 2, 2004.

[29] Cho, S.-H., and Eppinger, S.D., 2005, “A Simulation-Based Process Model for Managing Complex Design Projects,” IEEE Transactions on Engineering Management, **52**(3), pp. 316-328.

[30] Chang, T.-S., and Ward, A.C., 1995, “Conceptual Robustness in Simultaneous Engineering: a Formulation in Continuous Space,” Research in Engineering Design, **7**(2), pp. 67-85.

[31] Krishnan, V., D. Eppinger, S.D., and Whitney, D.E., 1992, “Ordering Cross-Functional Decision Making in Product Development,” working paper # 3299-91.

[32] Krishnan, V., Eppinger, S.D., and Whitney, D.E., 1997, “Simplifying Iterations in Cross-Functional Design Decision Making,” Journal of Mechanical Design, **119**, pp. 485-493.

[33] Shiau, Ching-Shin Norman, and Jeremy J. Michalek, “Should Designers Worry About Market Systems?” Journal of Mechanical Design, **131**(1), pp. 011011, 2009.

[34] Hazelrigg, G.A., 1998, “A framework for decision-based engineering design,” Journal of Mechanical Design, **120**, pp. 653-658.

[35] Renaud, J.E., and Gu, X., 2006, “Decision-Based Collaborative Optimization of Multidisciplinary Systems,” Decision Making in Engineering Design, K. Lewis, W. Chen, and L.C. Schmidt, eds., ASME Press, New York, pp. 173-186.

[36] Michalek, J.J., Ceryan, O., Papalambros, P.Y., and Koren, Y., 2006, “Balancing Marketing and Manufacturing Objectives in Product Line Design,” ASME Journal of Mechanical Design, **128**(6), pp. 1196–1204.

[37] Williams, N., Azarm, S., and Kannan, P.K., 2008, “Engineering Product Design Optimization for Retail Channel Acceptance,” ASME Journal of Mechanical Design, **130**(6), pp. 061402-1-10.

[38] Smith, P.G. and Reinersten, D.G., 1991, Developing Products in Half the Time, Van Norstrand Reinhold, New York.

[39] Karimian, Peyman, and Jeffrey W. Herrmann, 2009, “Separating Design Optimization Problems to Form Decision-Based Design Processes,” Journal of Mechanical Design, **131**(1), 011007.

[40] Gurnani, Ashwin, and Kemper Lewis, 2008, “Collaborative, Decentralized Engineering Design at the Edge of Rationality,” Journal of Mechanical Design, **130**(12), 121101.

[41] Chen, L., and Li, S., 2001, “Concurrent Parametric Design Using a Multifunctional Team Approach,” Proceedings of 2001 ASME International Design Engineering Technical Conferences and Computers and Information in Engineering Conference, DAC-21038, Pittsburgh, Pennsylvania, September 9-12, 2001.

[42] Simon, Herbert A., 1997, Administrative Behavior, fourth edition, The Free Press, New York.

[43] Gigerenzer, Gerd, Peter M. Todd, and the ABC Research Group, 1999, Simple Heuristics That Make Us Smart, Oxford University Press, New York.

[44] March, James, and Herbert Simon, 1993, Organizations, second edition, Blackwell, Cambridge, Massachusetts.

[45] Dym, Clive L., William H. Wood and Michael J. Scott, 2002, “Rank Ordering Engineering Designs: Pairwise Comparison Charts and Borda Counts,” Research in Engineering Design, **13**(4), pp. 236-242.

[46] Mintzberg, H., Raisinghani, D., and Theoret, A., 1976, “The Structure of Unstructured Decision Processes,” Administrative Science Quarterly, **21**(2), pp. 246-275.

[47] Nutt, Paul C., 2005, “Search During Decision-Making,” European Journal of Operational Research, **160**, pp. 851-876.

[48] Wang, Y., and Chiew, V., 2008, “On the Cognitive Process of Human Problem Solving,” Cognitive Systems Research, doi:10.1016/j.cogsys.2008.08.003.

[49] Simpson, T.W., 1998, “A Concept Exploration Method for Product Family Design,” Ph.D. thesis, Georgia Institute of Technology, Atlanta, Georgia.

[50] Wassenaar, Henk Jan, Deepak Kumar, and Wei Chen, 2006, “Discrete Choice Demand Modeling For Decision-

 Copyright © 2009 by ASME

Based Design," in Decision Making in Engineering Design, K. Lewis, W. Chen, and L.C. Schmidt, editors, ASME Press, New York.
[51] Brochtrup, Brad M., 2006, "Classifying and Comparing Design Optimization Problems," M.S. Thesis, University of Maryland, College Park.
[52] Wassenaar, H.J. and Chen, W., 2001. "An Approach to Decision-Based Design," DETC01/DTM-21683, Proceedings of DETC ASME Design Engineering Technical Conference, Pittsburgh, PA.
[53] Simon, Herbert A., 1981, The Sciences of the Artificial, second edition, MIT Press, Cambridge, Massachusetts.

Copyright © 2009 by ASME

ANNEX A

ENGINEERING PARAMETERS AND ATTRIBUTES

Engineering Parameters

Length of air gap l_g = 7.0 × 10-4 m

Terminal voltage V_t = 115 V

Resistivity of copper ρ = 1.69 × 10-8 $Ohms \bullet m$

Permeability of free space μ_o = 4 π × 10-7 H/m

Number of stator poles p_{st} = 2

Cost of copper C_c = 2.2051 $\$/kg$

Cost of steel C_s = 0.882 $\$/kg$

Density of copper δ_c = 8,960 kg/m^3

Density of steel δ_s = 7,861.09 kg/m^3

Relative permeability of steel [dimensionless] $\mu_{steel} = 1000$

Engineering Attributes

Magnetizing intensity [Ampere turns/ m]

$$H = N_c I / (l_c + l_r + 2l_g)$$

Mean path length within the stator [m]

$$l_c = \pi(2r_o + t_s)/2$$

Diameter of armature [m] $l_r = 2(r_o - t_s - l_g)$

Input power [W] $P_{in} = V_t I$

Power losses due to copper and brushes [W]

$$P_{out} = I^2(R_a + R_s) + 2I$$

Armature wire length [m] $l_{aw} = \left(2L + 2l_r\right) N_c$

Stator wire length [m] $l_{sw} = p_{st}(2L + 4(r_o - t_s))N_s$

Armature wire resistance [Ohm] $R_a = \rho l_{aw} / A_{aw} \times 10^6$

Stator wire resistance [Ohm] $R_s = \rho l_{sw} / A_{sw} \times 10^6$

Mass windings [kg] $M_w = (l_{aw} A_{aw} + l_{sw} A_{sw}) \delta_c \times 10^{-6}$

Mass of stator [kg] $M_s = \pi L(r_o^2 - (r_o - t_s)^2)\delta_s$

Mass of armature [kg] $M_a = \pi L(r_o - t_s - l_g)^2 \delta_s$

Motor constant [dimensionless] $K = N_c / \pi$

Magneto magnetic force [A turns] $\Im = N_s I$

Magnetic flux [Wb] $\varphi = \Im / \Re$

Total reluctance [A turns/ Wb] $\Re = \Re_s + \Re_a + 2\Re_g$

Stator reluctance [A turns/ Wb] $\Re_s = l_c / (2\mu_{steel}\mu_o A_s)$

Armature reluctance [A turns/ Wb]

$$\Re_a = l_r / (\mu_{steel}\mu_o A_a)$$

Reluctance of one air gap [A turns/ Wb]

$$\Re_g = l_g / (\mu_o A_g)$$

Cross sectional area of stator [m^2] $A_s = t_s L$

Cross sectional area of armature [m^2] $A_a = l_r L$

Cross sectional area of air gap [m^2] $A_g = l_r L$

Copyright © 2009 by ASME

ANNEX B

DERIVATION OF DEPENDENT DESIGN VARIABLES

First, we define the quantities Z and W, which depend upon the independent design variables, the engineering parameters, and the given torque:

$$Z = \frac{\pi(2r_0 + t_s)}{4\mu_{steel}\mu_0 t_s} + \frac{1}{\mu_{steel}\mu_0} + \frac{7\times10^{-4}}{\mu_0(r_0 - t - 7\times10^{-4})}$$
$$W = \frac{T\pi Z}{N_C N_S} \tag{B.1}$$

Because the total reluctance $\Re = Z / L$, the formula for torque can be rearranged to give the following relationship between L and I:

$$L = \frac{W}{I^2} \tag{B.2}$$

Using the relationships for P_{in} and P_{out}, we can derive the following expression:

$$P = 115I - \left(I^2(R_a + R_S) + 2I\right) \tag{B.3}$$

After including the definitions of R_a and R_S from Annex A and defining the quantities C_1, C_2, C_3, and C_4 (which also depend only upon the six independent design variables), we have an expression for power in terms of I and L:

$$C_1 = 1.69\times10^{-8} \times 10^{6} = 1.69\times10^{-2}$$
$$C_2 = \frac{2N_c}{A_{aw}} + \frac{4N_s}{A_{sw}}$$
$$C_3 = 4(r_0 - t - 7\times10^{-4})\frac{N_C}{A_{aw}} \tag{B.4}$$
$$C_4 = 8(r_0 - t)\frac{N_S}{A_{sw}}$$

$$P = 113I - C_1 I^2\left(C_2 L + C_3 + C_4\right) \tag{B.5}$$

Because P must equal 300, substituting Equation (B.2) into Equation (B.5) yields a quadratic function of I:

$$C_1\left(C_3 + C_4\right)I^2 - 113I + C_1 C_2 W + 300 = 0 \tag{B.6}$$

Copyright © 2009 by ASME