FUNCTIONAL THINKING IN COST ESTIMATION THROUGH THE TOOLS AND CONCEPTS OF AXIOMATIC DESIGN

by

LAEL ULAM ODHNER

SUBMITTED TO THE DEPARTMENT OF MECHANICAL ENFINEERING IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE DEGREE OF

BACHELOR OF SCIENCE
AT THE
MASSACHUSETTS INSTITUTE OF TECHNOLOGY

FEBRUARY 2004

Signature of Author..
Department of Mechanical Engineering
January 29, 2004

Certified by..
Nam P. Suh
Professor of Mechanical Engineering
Thesis Supervisor

Accepted by.......
Ernest G. Cravalho
Professor of Mechanical Engineering
Chairman, Undergraduate Thesis Committee

FUNCTIONAL THINKING IN COST ESTIMATION THROUGH THE TOOLS AND CONCEPTS OF AXIOMATIC DESIGN

by

LAEL ULAM ODHNER

Submitted to the Department of Mechanical Engineering
on January 29, 2004 in partial fulfillment of the
requirements for the degree of Bachelor of Science in
Mechanical Engineering

Abstract

There has been an increasing demand for cost estimation tools which aid in the reduction of system cost or the active consideration of cost as a design constraint. The existing tools are currently incapable of anticipating the unseen or latent effects of design changes made in an effort to cut cost. This paper presents an example of how the tools and concepts of axiomatic design theory can be integrated with the parametric cost estimation process, and then presents a series of arguments for why tools such as these which examine the functional architecture of a system are useful for optimizing cost at the preliminary design level.

Thesis Supervisor: Nam P. Suh
Title: Professor of Mechanical Engineering

Acknowledgements

This thesis is the result of months spent brainstorming cost estimation methods in an effort to introduce axiomatic design theory to cost estimators. This effort was undertaken primarily by Dr. Taesik Lee and Peter Jeziorek and overseen by Professor Nam P. Suh. It would like to thank all three for their great ideas, many of which are echoed in part or in full in this paper.

Table of Contents

Abstract 2
Acknowledgements 3
Table of Contents 3
1 Introduction 4
  1.1 Changing understandings of cost and cost estimation 4
  1.2 Current solutions to the cost estimation problem 4
  1.3 Why cost estimation needs improvement 5
  1.4 How to improve cost-conscious system design 5
2 A brief introduction to axiomatic design 7
  2.1 Defining a "system" in terms of functional requirements and constraints 7
  2.2 Design parameters and the mapping process 9
  2.3 Representations of design 10
  2.4 The axioms 11
  2.5 Further information on axiomatic design 14
3 Predicting cost and function in preliminary designs: an example of an integrated approach 15
  3.1 Contemporary practice: Parametric cost estimation 15
  3.2 A modeling method linking axiomatic design with parametric cost estimation 16
  3.3 Now that we have described the mapping, how is it useful? 20
4 Conclusion 26
  4.1 Summary 26
  4.2 Further directions 26
5 Works Cited 27

1 Introduction

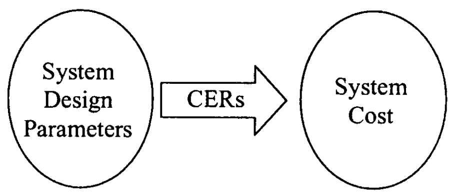
1.1 Changing understandings of cost and cost estimation

The role of cost in system design is changing significantly. Unlike the good old days of the space race and the arms race, when system performance was given a primary role in driving designs and budgets, cost is increasingly being seen as a controllable property of design decisions, rather than a static artifact of system design (SMAD 783). These new attitudes toward the role of cost, exemplified by NASA's "better, faster, cheaper" mantra, have redefined cost as a feature open to negotiation and optimization in the design of systems for space, defense, and other fields.

The increased focus on controlling cost has changed the goals of cost modeling. Whereas cost modeling was formerly used to produce one-off numbers for contract validation and budget predictions, there is now a need for costing tools which can serve to facilitate a running dialog between those responsible for the technical design decisions and those responsible for budgetary concerns. This new breed of tools must provide reliable results using automated methods to minimize the time and the number of people needed to produce an estimate.

1.2 Current solutions to the cost estimation problem

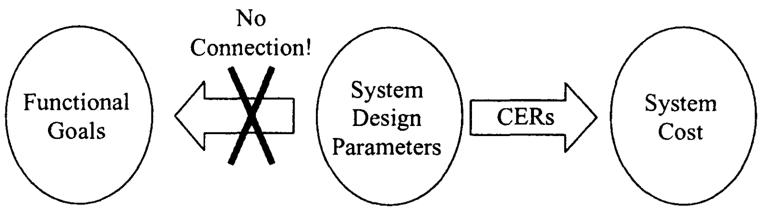
Government and industry experts have developed cost modeling systems which address the need for a fast, automated tool. Currently, the Department of Defense recommends that preliminary cost estimations should be based on the popular parametric cost estimation method (PCEH, 1999). In a parametric cost model, certain selected system design parameters are treated as inputs into cost estimating relationships (CERs) which yield a continuous range of cost estimates over a range of input values. The cost estimating relationships are determined from regression fits to historical cost data. When inflation and "learning curve" factors are taken into account, these cost estimating relationships do a reasonably good job at projecting the cost of designs which utilize well-established technologies, i.e. technologies for which credible historical data exists.



Because parametric cost estimates rely solely on statistical correlations, they have a great advantage over more rigorous cost methods which examine the cost of each component more thoroughly. Statistical cost estimating relationships require no expertise to use: anybody can read a design specification off of a drawing and plug it into a cost estimating relationship. Because of this, they are faster and cheaper than other methods (DOD letter to ISPA, 1999). The turnaround time from redesign to cost estimation is short enough that it is possible to go back and forth between financial constraints and design parameters in order to minimize cost.

1.3 Why cost estimation needs improvement

Despite these recent advances, current cost estimation tools provide only half of a solution to the problem of weighing system cost versus system performance. There are currently no good tools available to cost estimators for understanding the impact of design decisions on the ability of systems to satisfy their stated design goals. This lack of functional foresight could lead decision makers into a number of traps in an attempt to reduce cost.



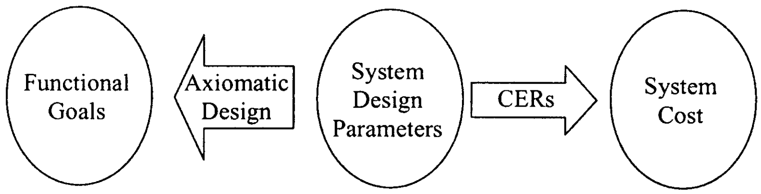
The primary risk of not directly examining the functional impact of some design variable is the psychological tendency to see cost reduction as a goal independent of the functional success of a design. Given a tool which can calculate with fairly high confidence the cost of a component, someone without knowledge of a component's functional impact might set about to cut cost and order a change which hurts system performance unacceptably. While such mistakes are generally caught in engineering design reviews, it is a better idea to avoid them altogether before time and money are spent on useless changes.

Another easy-to-make mistake is to cut back on one component to save money, only to spend money on other parts of the system to compensate for the reduced functionality of the cut part. The cost of compensation and the overhead incurred in making too many design changes is an easy way to turn what looks like a good cost opportunity into a host of problems further down the line, when the latent consequences of the design change emerge.

A lack of functional foresight could also cause cost-cutters to miss beneficial opportunities. Individual system components are often over-specified because their impact on the total system performance is overestimated. If such components can be identified and downgraded to components which still adequately serve their purposes, money can be saved without degrading functionality.

1.4 How to improve cost-conscious system design

The ideal cost optimization tool would be a bottom-up functional model of a system which allows designers to relate proposed technical changes to their implications in terms of both function and cost. This is clearly as impractical at the preliminary level as bottom-up cost estimation has proved to be, due to the amount of expertise and time which this level of modeling would require. However, the key features in the relationships between design parameters and performance can be analyzed using only simple engineering estimations and visualization tools. It would be extremely valuable just to have an understanding of which parameters affect which system functions, and the impact of each parameter relative to the others. This kind of information is easier to obtain and can provide a great deal of insight into where potential tradeoffs can be made in design, or where cost can be cut without affecting system performance at all.



The goal of this paper is to show that tools for the analysis of systems in this manner already exist within the framework of axiomatic design theory. The purpose of this paper is twofold. The primary goal is to promote the use of axiomatic design concepts such as the design matrix and functional-physical mappings as analytic tools to aid in the understanding of how to optimally design around cost constraints. The secondary goal is to provide an example of what a preliminary tool for analyzing systems in this manner might look like. The example given here is far from complete, but it shows how one can approach the problem of connecting cost with functional foresight at a preliminary design level. Some of the system analysis techniques presented in terms of axiomatic design go beyond this level, but the relevance of their concerns should not be lost on anyone who has tried to manage a project large enough to require proper project management.

Section two of this paper is an introduction to the concepts of axiomatic design theory. Those who are already familiar with axiomatic design may with to skip to section three, which demonstrates how axiomatic design can be used in the cost estimation process. Examples are given there of how both the first and second axioms of design can be used to choose how money is allocated in a preliminary design.

2 A brief introduction to axiomatic design

Axiomatic design is often treated purely as a design tool, rather than a tool for analysis. While it has been applied quite successfully as an automated and rigidly structured design process, the utility of the basic axiomatic design concepts extend beyond its invocation in this strict and all-encompassing manner. Axiomatic design theory presents a model for describing the nature of design and sets forth several criteria which define what a "good" design looks like within this model. The chief benefit of axiomatic design is that it presents systems in terms of relationships which are usually ignored, and which, if properly managed, can greatly improve the stability and ease of implementation of solutions to complex problems. As a tool for cost/function analysis, it is the quantification of these same relationships which make axiomatic design useful. This introduction to axiomatic design is consequently focused on introducing the representations of system design which axiomatic design uses and the core definitions which underlie them.

2.1 Defining a "system" in terms of functional requirements and constraints

The most crucial idea in axiomatic design theory is the definition of a system which serves as the foundation for axiomatic design's modeling tools. A system is a set of machines, procedures, and human operators which all combine to fulfill a set of *functional requirements* within a set of constraints. Functional requirements are the overarching criteria by which the success of a system can be defined. At the top level of a complex system, these can be fairly broad in scope; for example, a brief list of the functional requirements of an automobile might look something like this:

Example 1: top-level FRs of an automobile
FR_1: Transport 5 people and their luggage FR_2: Have a range of 600 miles on a 20 gallon tank of gas FR_3: Have a top speed of 90 MPH FR_4: Ensure passenger safety FR_5: Ensure passenger comfort

At the top level, the list of functional requirements (FRs) of any given system should encompass all of the system's goals in a manner akin to a mission statement, but in a structured and codified manner.

2.1.1 FRs are decomposable.

Within a single functional requirement, it should be possible to express a lower-level list of FRs which will result in the satisfaction of the parent FR. Often, this decomposition will require that some design decisions be made, as is the case with FR_1 from example 1. The requirement, "Transport 5 people and their luggage," could be met by providing seats for five and trunk space for five, but it would also be plausible to meet the requirement by providing five seats and under-seat storage space or roof rack space. Thus the sub-requirements could take either of these forms, depending on the designer's choice:

Example 1, continued: decomposition of FR_1
<u>With a roof rack:</u> $FR_{1.1}$: Provide 5 seats for passengers $FR_{1.2}$: Provide 10 cubic feet of trunk space $FR_{1.3}$: Provide structural support for a roof rack or car top carrier <u>Without a roof rack:</u> $FR_{1.1}$: Provide 5 seats for passengers $FR_{1.2}$: Provide 30 cubic feet of trunk space

Each sub-requirement in a set of FRs is also potentially decomposable into smaller FRs, so that the fleshed out specification for a system looks like a tree of requirements, as seen in Figure 2.1. This hierarchy provides a valuable tool not only for design, but for debugging, and will parallel the breakdown of function which one might expect in a failure modes effects analysis. Since the satisfaction of a parent FR is contingent on the satisfaction of all its sub-FRs, it is possible to isolate failure modes based on which FR is not satisfied at every level of requirement.

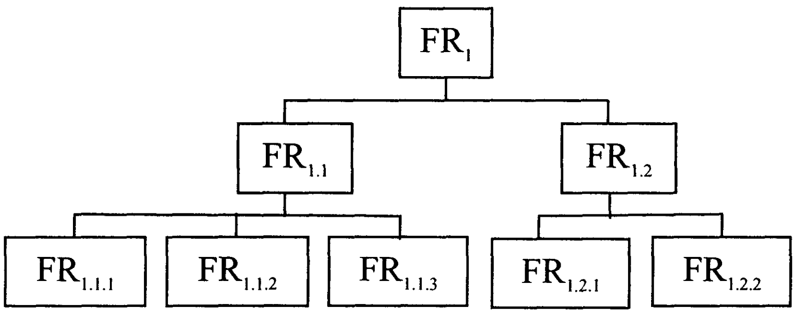


Figure 2.1: A functional hierarchy can be decomposed many times, depending on the level of granularity to which design decisions are made. The bottom or leaf-level functional requirements are measurable design criteria; in a proper hierarchy, the higher-level functional requirements are satisfied by their sub-FRs.

2.1.2 FRs are solution-neutral.

The individual mechanisms which are employed to satisfy a functional requirement are not important. For instance, it is immaterial to the driver of a car whether the car's electricity runs at 12 volts or at 42 volts, or whether the car's intake manifolds are made of metal or plastic. These details should not be prematurely specified, as that might constrain designers away from implementing the best solution possible. In some cases, artificial constraints are imposed in order to comply with industry standards, as is the case with the electricity in a car. These should be noted as constraints and enumerated separately so that they are well-understood and open to revision if they are deemed unnecessary at a future point.

2.1.3 FRs are quantifiable.

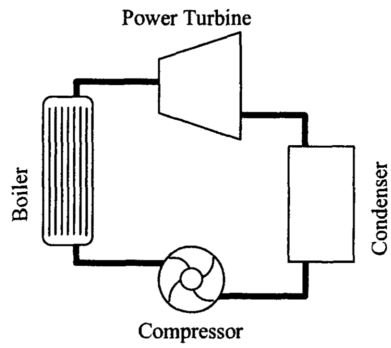
Finally, it is important that FRs are quantitative, not qualitative, whenever possible. The purpose behind stating the requirements of a system in a structured fashion is to give decision makers a solid contract which defines what a system should be capable of doing. Assertions such as “The force needed to mate the two connectors shall not exceed 70 Newtons” are ideal, because they provide a clear-cut numerical range over which the functional requirement is satisfied. Contracts of this kind are verifiable through mathematical modeling or testing. Within the larger context of this paper, it is easy to see how this is also important in assuring that the verification process can on some level be automated to provide an automated way of checking that the system requirements are satisfied.

2.2 Design parameters and the mapping process

2.2.1 Defining design parameters

According to axiomatic design theory, a proposed solution to a system design problem consists of two things: a set of adjustable design parameters (DPs) and information about how these design parameters affect the functional requirements of the system. The design parameters are the degrees of freedom which can be altered within the system structure in order to make the system work. In the simplest case, the DPs can be thought of as the dimensions on the blueprints of the system. However, just like FRs, DPs can be decomposed to multiple levels of design according to the level of detail which is required. In a preliminary design, for instance, a system’s design parameters might be high-level component specifications, as is shown in example 2. The Rankine cycle generator schematic of example 2 does not need to be defined at the level of the dimensions or materials of each component; the system performance can be assessed from the parameters given.

Example 2: The high-level design parameters of a Rankine steam generator



Compressor
Compressor output pressure
Compressor throughput
Compressor efficiency

Boiler
Boiler efficiency
Boiler output temperature

Power Turbine
Turbine efficiency
Turbine expansion ratio

Condenser
Condenser NTU
Condenser efficiency

2.2.2 How design parameter map onto functional requirements

The other component of design solution is the schematic relationship that defines how the parameters impact the functional requirements of the system. This relationship is embodied in the flowcharts, block diagrams, and blueprints of the design. In a well-described design, all of the design parameters form a state space which determines whether or not the functional requirements of the system are satisfied. Each functional requirement can be thought of as an equation of the form $FR_{i,\min} < f_i(DP_1, DP_2, \dots DP_m) < FR_{i,\max}$, where $FR_{i,min}$ and $FR_{i,max}$ are lower and upper bounds on the numerical range over which the FR_i is satisfied.

The functions which map the DPs onto the functional requirements are performance metrics determined from engineering analysis

2.2.3 Finding the correct set of design parameters

The mappings from design parameters onto functional requirements combine to form a set of simultaneous equations that defines the complete solution to the system:

$$
\begin{gathered}
FR_{1,\min} < f_1(DP_1, DP_2, \ldots DP_m) < FR_{1,\max} \\
FR_{2,\min} < f_2(DP_1, DP_2, \ldots DP_m) < FR_{2,\max} \\
\vdots \\
FR_{n,\min} < f_n(DP_1, DP_2, \ldots DP_m) < FR_{n,\max}
\end{gathered}
$$

System designers spent a lot of time and money trying to come up with a set of design parameters which satisfy these equations. As the number of functional requirements increases, the number of equations increases, and this can make it seem very hard to solve for the DPs of a large system. A few tools are needed to make the solving simpler.

2.3 Representations of design

It is fortunate that most systems are inherently modular in the sense that design parameters have a localized effect on the system functional requirements. If the nature of this localization is understood, the process of solving the system of functional requirements can become much easier. A complex system might have thousands of design parameters even at a relatively high level, but if most of these can be ignored for each specific functional requirement, the process is actually quite manageable. To this end, several simplified representations of system design have been invented within the framework of axiomatic design. These are the coupling and stiffness matrices.

2.3.1 The coupling matrix

It is often useful to simply consider whether a DP affects a given FR at all. The coupling matrix is a matrix mapping the DPs onto the FRs which contains an X for every element where the DP affects the FR, and a 0 for every element where a FR is independent of the DP.

$$
\begin{Bmatrix} FR_1 \\ FR_2 \\ FR_3 \\ \vdots \\ FR_n \end{Bmatrix} = \begin{bmatrix} X & 0 & X & \cdots & 0 \\ 0 & X & 0 & \cdots & X \\ 0 & X & X & \cdots & 0 \\ \vdots & \vdots & \vdots & & \vdots \\ X & 0 & X & \cdots & X \end{bmatrix} \begin{Bmatrix} DP_1 \\ DP_2 \\ DP_3 \\ \vdots \\ DP_m \end{Bmatrix}
$$

Figure 2.2: The coupling matrix indicates the presence or absence of coupling between a functional requirement and a design parameter.

2.3.2 The stiffness matrix

The stiffness matrix goes a step beyond the coupling matrix to indicate not only the dependence of a functional requirement on a design parameter, but also the degree of this dependency. It can be thought of as a linearization of the functional requirements about some point in the design parameter space which is close to the solution. It is useful because it can be used to isolate the design parameters which have the highest impact on a particular functional requirement and thus are the easiest points to impact the value of the functional requirement, and also those which are potentially too sensitive and thus sources of unwanted variation.

$$\begin{Bmatrix} \Delta FR_1 \\ \Delta FR_2 \\ \Delta FR_3 \\ \vdots \\ \Delta FR_n \end{Bmatrix} = \begin{bmatrix} \frac{\partial FR_1}{\partial DP_1} & \frac{\partial FR_1}{\partial DP_2} & \frac{\partial FR_1}{\partial DP_3} & \cdots & \frac{\partial FR_1}{\partial DP_m} \\ \frac{\partial FR_2}{\partial DP_1} & \frac{\partial FR_2}{\partial DP_2} & \frac{\partial FR_2}{\partial DP_3} & \cdots & \frac{\partial FR_2}{\partial DP_m} \\ \frac{\partial FR_3}{\partial DP_1} & \frac{\partial FR_3}{\partial DP_2} & \frac{\partial FR_3}{\partial DP_3} & \cdots & \frac{\partial FR_3}{\partial DP_m} \\ \vdots & \vdots & \vdots & & \vdots \\ \frac{\partial FR_n}{\partial DP_1} & \frac{\partial FR_n}{\partial DP_2} & \frac{\partial FR_n}{\partial DP_3} & \cdots & \frac{\partial FR_n}{\partial DP_m} \end{bmatrix} \begin{Bmatrix} \Delta DP_1 \\ \Delta DP_2 \\ \Delta DP_3 \\ \vdots \\ \Delta DP_m \end{Bmatrix}$$

Figure 2.3: The stiffness matrix in its abstract form can be thought of as the partial derivative of each functional requirement with respect to each design parameter.

2.4 The axioms

All of this terminology which has just been introduced—functional requirement trees, design parameters, coupling and stiffness matrices—is the language in which design is described by axiomatic design theory. What makes a design good or bad according to axiomatic design is determined by the axioms of design, which are as follows:

1. The Independence Axiom: Each design parameter should affect only one functional requirement, so that the functional requirements are all independent of one another.
2. The Information Axiom: Each functional requirement should be satisfied robustly; that is, with 100% probability despite variation of the design parameters.

Both of these axioms make sense from a common-sense design perspective. What makes them so powerful is that the tools provided by axiomatic design allow for the examination of these assertions in a rigorous, quantified, and possibly automated manner. Below is an in-depth description each axiom.

2.4.1 The independence axiom

Most design changes have both intended and unintended consequences when design parameters affect more than one functional requirement of a system. Because it is necessary to compensate for these unintended consequences of parameter variation, one innocuous design change can spiral into a multitude of adjustments in order to keep all of the system's functional requirements satisfied at once. The solution to this problem is to avoid "coupling" two functional requirements together in this way. Axiomatic design recognizes three basic kinds of coupling: uncoupled design, decoupled design, and fully coupled design.

2.4.1.1 Uncoupled design

Ideally, no system design parameter would affect more than one functional requirement. This is called uncoupled design, and is represented by a coupling matrix which has no elements off of the diagonal:

$$\begin{Bmatrix} FR_1 \\ FR_2 \\ FR_3 \\ FR_4 \end{Bmatrix} = \begin{bmatrix} X & 0 & 0 & 0 \\ 0 & X & 0 & 0 \\ 0 & 0 & X & 0 \\ 0 & 0 & 0 & X \end{bmatrix} \begin{Bmatrix} DP_1 \\ DP_2 \\ DP_3 \\ DP_4 \end{Bmatrix}$$

This system is the easiest kind to adjust because each of the equations which make up the functional requirements is independent of the others, and consequently a change in any requirement will not demand the re-examination of any equations other than the one which is affected.

2.4.1.2 Decoupled design

A design is considered to be decoupled when the functional requirement can be ordered in a manner that allows for the sequential solution of all of the functional requirements without any backtracking. Consider the following coupling matrix:

$$\begin{Bmatrix} FR_1 \\ FR_2 \\ FR_3 \\ FR_4 \end{Bmatrix} = \begin{bmatrix} X & 0 & 0 & 0 \\ X & X & 0 & 0 \\ X & 0 & X & 0 \\ 0 & 0 & X & X \end{bmatrix} \begin{Bmatrix} DP_1 \\ DP_2 \\ DP_3 \\ DP_4 \end{Bmatrix}$$

If the equations are solved starting with FR_1 and proceeding down the list, no iteration is required to solve this system. DP_2 can be varied to solve FR_2, DP_3 can be varied to solve FR_3, and so on. The key feature of the coupling matrix which indicates that a system is decoupled is that the off-diagonal dependencies are all below the diagonal, so that no parameter adjusted will feed back into the already adjusted parameters above it.

2.4.1.3 Fully coupled design

When the design parameters are so thoroughly interrelated through the functional requirements that there is no alternative to solving them simultaneously, a system is considered to be fully coupled. This state of affairs is very difficult because the only way to adjust the parameters of the system is through painstaking modeling or iterative trial and error. When represented as a coupling matrix, a fully coupled system has off-diagonal dependencies above the diagonal as well as below, no matter how the functional requirements are ordered:

$$\begin{Bmatrix} FR_1 \\ FR_2 \\ FR_3 \\ FR_4 \end{Bmatrix} = \begin{bmatrix} X & 0 & X & 0 \\ 0 & X & X & X \\ X & 0 & X & 0 \\ X & X & 0 & X \end{bmatrix} \begin{Bmatrix} DP_1 \\ DP_2 \\ DP_3 \\ DP_4 \end{Bmatrix}$$

The best that one can do when faced with a fully coupled system that is unavoidable is to limit the coupling to as few elements as possible. Solving for two or three design parameters simultaneously may not be too hard; solving for more will get quite complicated.

2.4.2 The information axiom

Any good designer knows that variation is unavoidable in real-world components. Every dimension and property given has a tolerance and a statistical distribution. These DP variations propagate through the system design to become variations in function, which may or may not be problematic. If all of the variation lies within the acceptable design range, as is shown in Figure 2.4, the system is still reliable and robust. However, if the net variation in the output functional requirement is large enough to exceed the width of the required design range, or if the distribution of the functional metric is skewed away from the center of the design range, the resulting error will propagate up through the functional hierarchy and cause a high-level system failure.

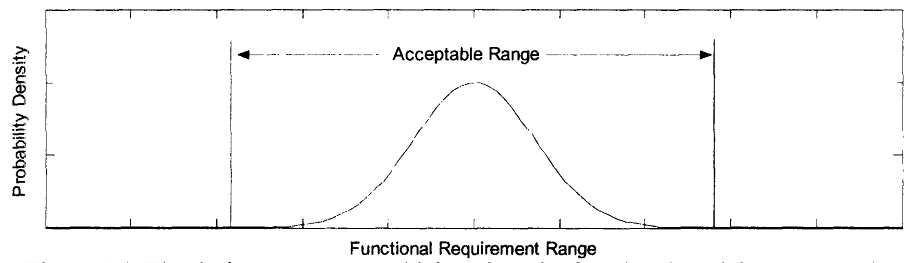


Figure 2.4: The design parameters which go into the functional models vary, creating a probabilistically distributed estimate of the value of the functional metric. A functional requirement is considered satisfied by the information axiom if and only if the integrated area under the probability distribution function is 1 between the lower and upper bounds of the functional requirement.

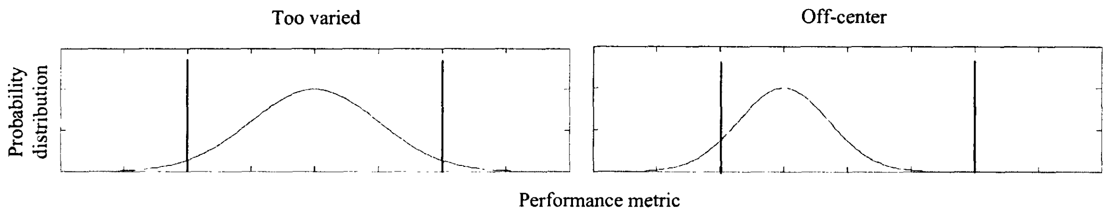


Figure 2.5: There are two ways in which parameter variation can cause a functional requirement to fail; the first is too much variation in the inputs (shown left), and the second is improper "centering" of the probability distribution function within the functional requirement's design range (shown right).

The elements of the stiffness matrix can be used to roughly calculate the expected variation in a functional requirement if the variation in the inputs is assumed to be Gaussian. For a first-order Taylor expansion in several variables, the net standard deviation of the output can be written in terms of the weighted sum of the squares of the individual variables, i.e. if $x_{total} = a_1x_1 + a_2x_2 + a_3x_3 + \ldots + a_nx_n$, then $\sigma_{total}^2 = a_1^2\sigma_1^2 + a_2^2\sigma_2^2 + a_3^2\sigma_3^2 + \ldots + a_n^2\sigma_n^2$. In terms of the stiffness matrix, the standard deviation for any FR can be estimated as $\sigma_{FR,i}^2 = \sum_j \left(\frac{\partial FR_i}{\partial DP_j} \right)^2 \sigma_{DP,j}^2$ (Suh, p.75).

Since the partial derivative terms in this expression are contained in the row vectors of the stiffness matrix, all of the standard deviations could be expressed using a matrix like the stiffness matrix, but with all the elements squared:

$$\begin{Bmatrix} \sigma_{FR,1}^2 \\ \sigma_{FR,2}^2 \\ \vdots \\ \sigma_{FR,n}^2 \end{Bmatrix} = \begin{bmatrix} S_{11}^2 & S_{12}^2 & \cdots & S_{1m}^2 \\ S_{21}^2 & S_{22}^2 & \cdots & S_{2m}^2 \\ \vdots & \vdots & & \vdots \\ S_{n1}^2 & S_{n2}^2 & \cdots & S_{nm}^2 \end{bmatrix} \begin{Bmatrix} \sigma_{DP,1}^2 \\ \sigma_{DP,2}^2 \\ \vdots \\ \sigma_{DP,m}^2 \end{Bmatrix}$$

From the expression above it is easy to see that a DP which has a large impact on a given FR has the potential to introduce a great deal of variation into that FR. Consequently, a corollary to the information axiom is that stiffness should be reduced in functional requirements which are particularly variation-sensitive, or in design parameters which are particularly noisy. The alternative approach, to tighten the tolerance on a design parameter, is expensive and better avoided if possible.

Additionally, it is easy to see from the stiffness matrix that the number of couplings between design parameters and functional requirements plays a large role in the amount of variation present. Each coupling adds variation to the functional requirements; in this way the information axiom echoes the independence axiom.

2.5 Further information on axiomatic design

Many good articles and several books have been written arguing and illustrating how design according to the independence and information axioms leads to robust, adjustable systems. For the sake of space, those arguments will not be repeated here. Additional information about axiomatic design can be found in Axiomatic Design: Advances and Applications by Nam P. Suh. The remainder of this paper instead focuses on the application of these basic axioms and representations to the task of evaluating and reducing the cost of a system.

3 Predicting cost and function in preliminary designs: an example of an integrated approach

3.1 Contemporary practice: Parametric cost estimation

3.1.1 Why build on parametric cost estimation?

I have chosen parametric cost estimation as a starting point for improving cost estimation because it is primarily useful in the conceptual design stages of a project. This is also a stage in which basic architecture decisions (and mistakes) are being made, so axiomatic design can potentially be useful to evaluate design as well as optimize cost. Furthermore, both of these tools seek to understand the implications of parametric variation in design, so integrating the two tools into one way of looking at design is not a difficult task.

3.1.2 How parametric cost estimation is performed

The basic approach of parametric cost estimation is to identify the breakdown of subsystems and components within a project, to extract the cost-relevant parameters from these components, and then to apply cost estimating relationships (CERs) to predict the various costs associated with each one. These cost estimating relationships are regression-fit models which take as inputs one or two parameters such as weight, power output, or accuracy, as the example values in Figure 3.1 show. The cost values produced by cost estimating relationships tend to be in units such as thousands of dollars as valued in the year 2000, or other similarly inflation-independent terms.

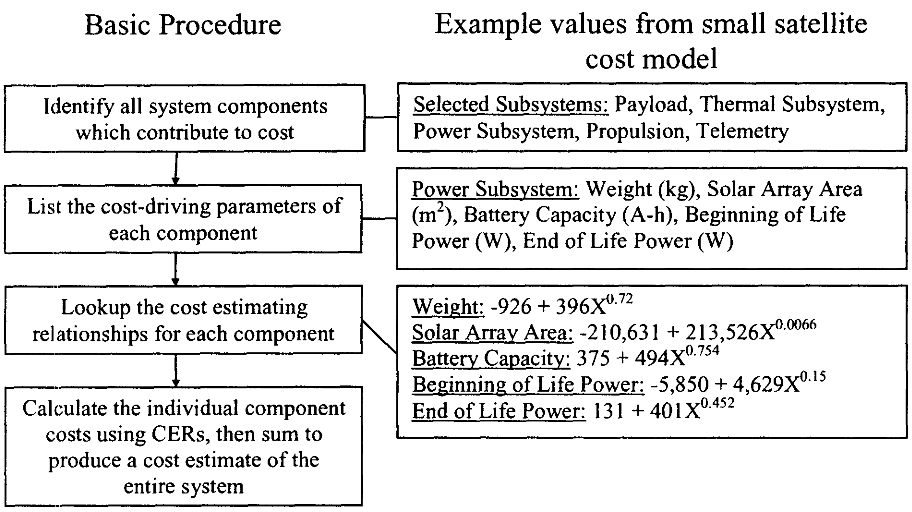


Figure 3.1: Typical process for creating a parametric cost estimate (summarized from table 20-2, SMAD, p. 792)

3.2 A modeling method linking axiomatic design with parametric cost estimation

Figure 3.2 shows one possible set of steps through which axiomatic design concepts and parametric cost estimating relationships can be applied at a preliminary level to obtain an understanding of how cost and function interact. It is basically an attempt to fuse together the concepts and methods of axiomatic design with the concepts and methods of parametric cost estimation. In this method, the analysis is begun just as in any axiomatic design project, with a functional breakdown and with an enumeration of the design parameters of the system at a high level. These design parameters are then related to the functional requirement through basic performance metrics obtained by engineering analysis. The fusion process really begins when the design parameters and the cost-driving variables which are used as inputs by the CERs must be related to each other, which is done by writing the CER inputs in terms of the design parameters, since the design parameters are more likely to be definable by the analyst and thus can be made more flexible. Once it is possible to model the system cost through CERs, basic cost optimization can be performed by examining the system architecture for coupling and other large-scale problems, finding the parameters which can be trimmed due to excessive safety margins, and identifying possible tradeoffs between design parameters which will yield a working and cheaper system.

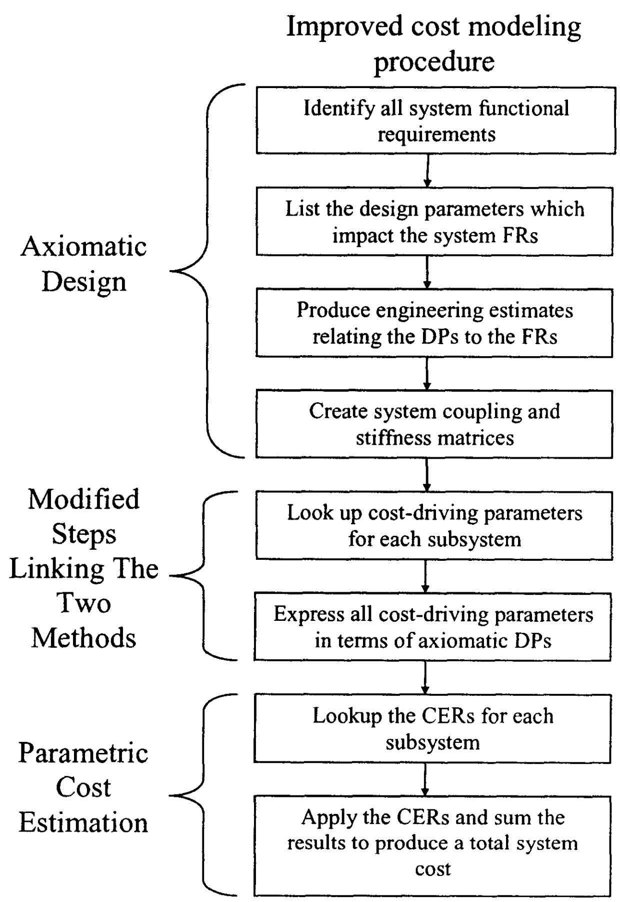


Figure 3.2: The proposed method of cost estimation

Included with this method is a brief example of what the process of analyzing a system in this way might look like. Although there were no cost estimating relationships available to give accurate numbers, the basic gist of the process is easier to follow with an example. This example, illustrated in Figure 3.3, elaborates on example 2 from the earlier section, and is a partial high-level schematic for a steam power plant.

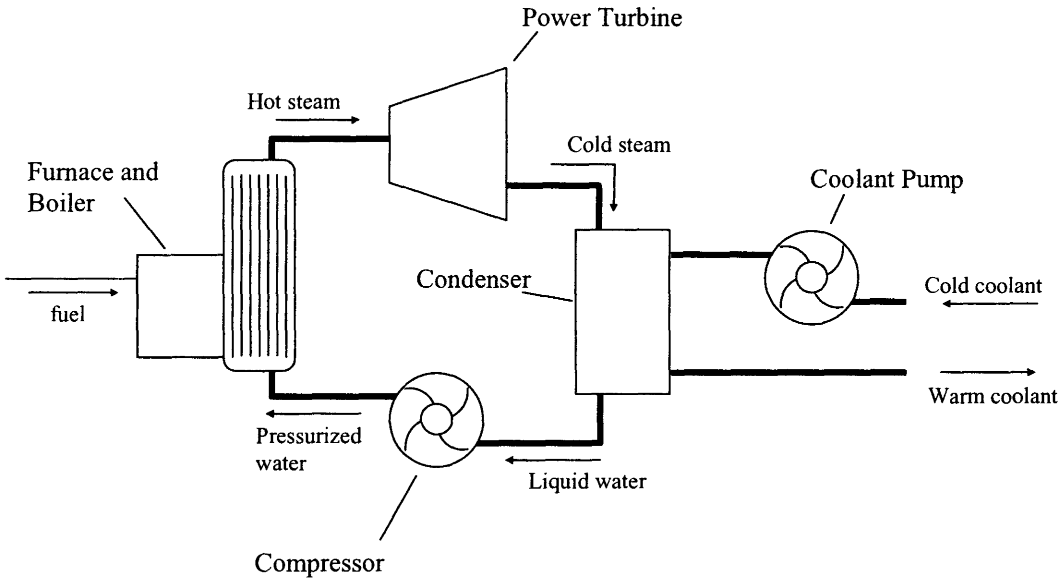


Figure 3.3: The block diagram for a simple Rankine cycle power plant.

3.2.1 Determining the functional decomposition of the system

The first step in producing an interactive cost model which examines functionality is to ascertain what exactly the functional requirements of the system are. If these requirements have not been established at the time of design, they can be established retrospectively, though care must be taken not to overlook anything which might be an unstated goal. The basic breakdown of the functional requirements should continue to the level where the design parameters match the level of design detail which is shown on the preliminary design. In the example at hand, a basic set of functional requirements for the system probably look like this:

FR_1: Produce X megawatts of power.
FR_2: Require a fuel input of no more than Y megawatts.
FR_3: Require a compressor input of no more than Z megawatts.
FR_4: Let waste heat into the environment at a temperature of no more than 35 C.

3.2.2 Enumerating all of the system design parameters

Once a level of decomposition commensurate with the level of design detail has been reached, all of the design parameters which affect each one of the functional requirements must be catalogued. Unlike a traditional axiomatic design approach, in which one design parameter is chosen to be varied for each functional requirement, all of the design parameters in the system must be taken into account here, since all of them potentially vary and also potentially affect cost. Table 3.1 lists the design parameters from the example, omitting parameters which are constrained by the basic system structure (i.e. turbine throughput = pump throughput by conservation of mass).

Table 3.1: Design parameters of a Rankine cycle generator		
Component	Parameter	Variable Name
Boiler	Boiler efficiency	ε_b
	Boiler outlet temperature, K	T_h
Turbine	Turbine efficiency	ε_t
	Turbine expansion ratio	R
Condenser	Condenser efficiency	ε_{he}
	Condenser NTU	NTU
Compressor	Compressor throughput, kg/s	M_w
	Compressor efficiency	ε_{comp}
	Compressor output pressure, kPa	P_h
Coolant pump	Coolant flow rate, kg/s	M_{cool}

3.2.3 Producing engineering estimates

Like any ordinary axiomatic design problem, the next step is to produce a set of estimates which predict the impact of each DP on each FR. The point of this exercise is not to produce any kind of detailed analysis beyond what it should take an engineer several hours to do with pencil and paper. Instead, the purpose is to act as a good bounding case so that there can be some kind of "sanity check" on the parameters as they are varied. Below are the performance metrics for the example FRs:

FR_1:

Output power $= W_{out} = \varepsilon_c \cdot M_w \cdot \left(h_{H_2O}(P_h, v_h) - h_{H_2O}(P_l, v_l)\right)$

h_{H_2O} is a lookup function for the enthalpy of supersaturated steam

v_h, the specific volume of the hot steam, is found on a steam table using P_h and T_h

P_l, the output pressure of the turbine, is given by the relation $P_l = P_h / R$

v_l, the reversible specific volume of the steam at the output pressure, is found using a lookup table.

FR_2:

Fuel required $Q_{fuel} = \dfrac{M_w \cdot c_{p,H_2O} \cdot (T_h - T_{cond})}{\varepsilon_b}$

T_{cond}, the condenser outlet temperature, is a function ε_{cond}, NTU, and the coolant inlet temperature.

FR_3:

Compressor input power $W_{comp} = \dfrac{M_w \cdot (P_h - P_l)}{\rho_{H_2O} \cdot \varepsilon_c}$

FR_4:

Waste heat temperature $T_{output} \approx T_{coolant,in} + \frac{M_w}{M_{cool} \cdot c_{p,H_2O}} \cdot h_{fg,H_2O}\Big|_{P_l}$ (Assuming that the steam is not very superheated coming out of the turbine)

From these estimates, the coupling and the relative stiffnesses of the DPs can be obtained for each FR, and the matrices can be written out to visualize these relationships. Up to this point the analysis is basically the same kind of analysis which one would use to evaluate any preliminary design problem. Now the key is to connect this to the cost estimation problem.

3.2.4 Mapping system DPs onto cost-driving parameters

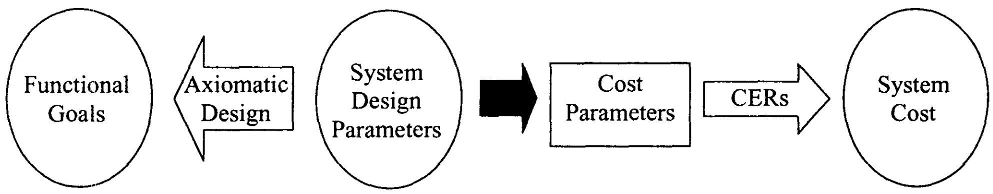


Figure 3.3: An intermediate step is often needed to determine values for parameters with significant impact on cost but no useful meaning in terms of function.

The parameters of a component which drive the cost are often very different from the functional design parameters. Parameters which usually have very little functional significance, such as weight, often correlate very well with cost. In order for the connections between cost and function to be made, designers must establish some kind of relationship between the DPs of axiomatic design and the DPs. This is possible by establishing a set of functions which map DPs onto cost-driving parameters. In situations where the cost-driving parameter is largely linked to a functionally significant quantity, such as battery life, solar cell array size, or radio transmitter strength, the mapping might be as simple as passing a DP through to the cost estimating relationships. To map DPs onto quantities such as weight, it may be necessary to resort to more crude methods such as looking up values in tables or using dimensions and density to produce rough estimates.

In the steam generator example, let's say that the condenser is basically a cross-flow heat exchanger and that the cost of the heat exchanger has been determined to be a function of the net length of piping used. The relationship between the length of piping used in the heat exchanger and number of transfer units (NTU) of the heat exchanger is given by the definition of NTU. $NTU = U \cdot A / C_{\min}$, where U is the bulk heat transfer coefficient of the heat exchanger, A is the total area of the heat exchanger, and C_{min} is the minimum heat capacity of the flow, taken in this case to be the coolant flow rate M_{cool} multiplied by the heat capacity of water. Writing A in terms of the surface area of the pipes,

$$NTU = U \cdot \pi \cdot D \cdot L \cdot \frac{1}{C_{\min}} = \frac{U \cdot \pi \cdot D \cdot L}{C_{\min}} \Rightarrow L = \frac{NTU \cdot C_{\min}}{U \cdot \pi \cdot D}$$

L in this equation is the desired quantity, the net length of the heat exchanger tubing. The bulk heat transfer coefficient used could either be determined from a series of Nusslet number correlations or empirically estimated from the performance specifications from a number of potential heat exchangers. Not all mappings will be so susceptible to estimation, and it is probable that a truly solid set of CERs would simply have to be defined in more functional units.

Some cost-driving parameters will be more closely linked to the tolerance of a particular DP than to its magnitude. There are certainly simple cases which confirm this, like that of a steel bar cut to a length of a foot and a tolerance of $1/1000^{th}$ of an inch. Were the bar one and a half feet long, the cost would probably still be more closely related to the cost of the machining operation needed to obtain that precision than to the cost of the material. This is relevant to cost estimation because it highlights a key difference between the ways in which DPs and inputs to CERs are defined.

3.3 Now that we have described the mapping, how is it useful?

The relationship between function and cost which has been developed is useful mainly in terms of the two axioms of design: independence and information. The independence axiom predicts the relative ease with which a design can be changed, and thus has application when several cost-cutting changes are being weighed against one another. The information axiom allows designers to see how much a design parameter can be changed and still satisfy all system functional requirements.

3.3.1 Coupling and cost

The first axiom of design states that the amount of coupling in design should be minimized. This makes sense from a cost perspective as well as from a reliability perspective. Coupling is expensive in several ways. Because altering a coupled DP necessitates the readjustment of other DPs in order to compensate for its affect on other FRs, the cost of many components could change as a result of changing just one. The overhead cost incurred in making design changes also adds up when many changes have to be made. And most importantly, an unanticipated coupling can cause FRs to fail unexpectedly. If a problem of this sort is not caught until late in a program's development, the price increases substantially.

3.3.1.1 Quantifying the propagation of design changes

The coupling matrix can be used to quantify the degree to which a design parameter is coupled, aiding in the selection of DPs which are easy to modify. The simplest approach is to only adjust components which are not coupled to multiple functional requirements. Figure 3.5 shows how examining the column vectors of the coupling matrix can indicate the number of FRs directly impacted by each DP. The immediate impact of a design change in an uncoupled component is easy to assess because it does not propagate to other functional requirements or through other design parameters. This does not mean that the tradeoff between the cost and function of that DP is potentially any greater than other possible DPs, but it does mean that the tradeoff is clearly observable.

$$
\begin{array}{ccccccc}
 & & \text{Bad to adjust} & & \text{Good to adjust} & & \\
\begin{Bmatrix} FR_1 \\ FR_2 \\ FR_3 \\ FR_4 \\ FR_5 \end{Bmatrix} & = & \multicolumn{4}{c}{\begin{bmatrix} X & 0 & 0 & 0 & 0 \\ X & X & 0 & 0 & 0 \\ X & 0 & X & 0 & 0 \\ 0 & 0 & 0 & X & 0 \\ 0 & X & 0 & X & X \end{bmatrix}} & \begin{Bmatrix} DP_1 \\ DP_2 \\ DP_3 \\ DP_4 \\ DP_5 \end{Bmatrix}
\end{array}
$$

Figure 3.5: Uncoupled DPs, such as DP_3 in the matrix above, are often simpler to adjust in a cost-saving effort because their effects are visible in only one functional requirement.

If it is necessary to adjust a parameter which is coupled to others through multiple FRs, care must be taken to consider all of the potential changes which must be made to compensate. In a decoupled matrix such as the one shown in Figure 3.6, a change made to a DP which is close to the top of the matrix can propagate downward quite a distance, while a change made near the bottom of the matrix necessitates far fewer alterations. Fortunately, the coupling matrix provides enough information to enumerate the set of DPs which are coupled to any particular one. The procedure is as follows:

1. List the FRs which contain an X in the column corresponding to the DP.
2. For each of these FRs, list the DPs which contain an X in the row corresponding to the FR.
3. Recursively apply this procedure to each of the DPs in the list until all possibly affected DPs are discovered.

Careful readers might note that this procedure can cycle indefinitely if the matrix being analyzed is fully coupled. While it is easy to detect this for the purposes of discovering coupled DPs, it hints at a bigger problem with fully coupled designs. When adjusting the parameters of a fully coupled system, it will be necessary to either solve several FRs simultaneously, or to iteratively make design changes until all of the FRs are satisfied. At the preliminary design stage, this might not seem so bad, because there is relatively little overhead associated with making parameter changes. However, while the parametric cost estimations might not predict a large rise in cost due to this kind of coupling, the overhead at later stages in project development will become significant. If a fully-coupled set of parameters is detected, the basic system architecture should be reexamined during the preliminary stages of design so that these problems can be avoided further down the road.

DP_1 affects both FR_1 and FR_2

DP_2 affects both FR_2 and FR_3

And so on…

$$\begin{Bmatrix} FR_1 \\ FR_2 \\ FR_3 \\ FR_4 \\ FR_5 \end{Bmatrix} = \begin{bmatrix} X & 0 & 0 & 0 & 0 \\ X & X & 0 & 0 & 0 \\ 0 & X & X & 0 & 0 \\ 0 & 0 & X & X & 0 \\ 0 & 0 & 0 & X & X \end{bmatrix} \begin{Bmatrix} DP_1 \\ DP_2 \\ DP_3 \\ DP_4 \\ DP_5 \end{Bmatrix}$$

Figure 3.6: Despite the fact that each of the first four design parameters affect two DPs directly, adjusting the first implies changes to five DPs, whereas adjusting the fourth implies adjusting only two.

Parameter adjustments can get caught in an infinite loop

$$\begin{Bmatrix} FR_1 \\ FR_2 \\ FR_3 \\ FR_4 \\ FR_5 \end{Bmatrix} = \begin{bmatrix} X & 0 & X & 0 & 0 \\ 0 & X & 0 & 0 & 0 \\ X & 0 & X & 0 & 0 \\ 0 & 0 & 0 & X & 0 \\ 0 & 0 & 0 & X & X \end{bmatrix} \begin{Bmatrix} DP_1 \\ DP_2 \\ DP_3 \\ DP_4 \\ DP_5 \end{Bmatrix}$$

Figure 3.7: Parameter adjustments caught in an infinite loop will make iteratively applied design changes very expensive in terms of overhead.

3.3.2 Information and cost

For the purposes of cost-constrained design, information axiom helps understand how over-engineered a system is. Including a generous margin in performance specifications is good engineering practice, but if cost is a concern, it is also a potential waste. If the risks associated with a particular functional requirement are well-understood, there is no reason why a system should be substantially over-specified. This includes situations in which the magnitude of a performance metric far exceeds the required value, and also situations in which the tolerance on a functional requirement metric is far tighter than necessary.

3.3.2.1 Safety factor over-engineering

One great fallacy of engineering projects is that "taking a safe value and doubling it" is always necessary. In Figure 3.8, the top plot shows a functional requirement which is far exceeded by system performance. Assuming that the estimate or bounding case for the probability distribution of the functional requirement is resonable, the mean performance could be reduced significantly with respect to this functional requirement and the system would still maintain a 0% failure probability. The engineering estimates of function are useful in this regard because they allow the designer to interactively adjust both the functional requirements of the system and the cost by varying the system DPs. If the tolerance of the DPs is driving cost, rather than the magnitude of the DPs, cost might be more effectively cut by allowing the variation of performance to increase.

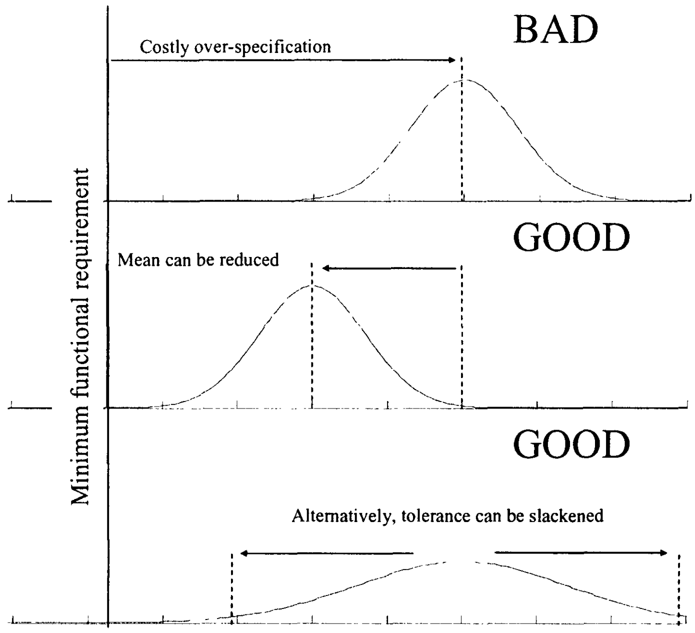


Figure 3.8: An over-engineered system which exceeds its minimum performance level can be made cheaper by either cutting DPs to reduce the mean value of the performance or by relaxing the control over the DP tolerances. As long as the area under the probability distribution function within the allowable performance range is still equal to 1, there is no disadvantage to either of these strategies.

A good example of safety factor over-engineering that can be corrected would be the over-specification of a radio transmitter for use with a planetary rover. Assume that the functional requirement is that the data transmission rate of the rover must exceed some minimum value during normal operation. The design parameters which impact this might be the power capacity of the electrical power subsystem, the transmitter strength, the signal processing equipment, and the antenna, among others. Most likely, the functional metric would look something like $D = W \log (G*T/N)$, where D is the data rate, W is the bandwidth of the signal processing equipment, G is the antenna gain, T is the transmitter power, and N is the expected background noise. An associated and coupled FR in the power subsystem would be that the power system be capable of providing some power P to the transmitter which is governed by the relationship $P = 2*T$. Assuming that the current data rate as design is far larger than necessary, the opportunities for cutting cost are either to reduce the bandwidth of the signal processing system, or to reduce the transmission power. Reducing the transmission power seems like a better idea for several reasons. First, data rate varies logarithmically with the radiated antenna power, so a significant cut in power yields a smaller cut in the performance metric than a cut in bandwidth might. Furthermore, due to the coupling between T and P, the reduction of the transmission power would lead to a decrease in the power requirement for the power subsystem, leading to further cost reductions.

3.3.2.2 Over-tolerancing

When a functional requirement asserts that the value of some performance metric must lie between two values, as in the case of a thermal control system which must be able to keep temperature between two values, another cost-saving opportunity arises. Figure 3.9 illustrates this scenario, and shows how the performance of a system can exceed its specified values by lying too far within a range. The stiffness matrix is helpful in finding out how far a particular tolerance can be relaxed in order to keep the system within its needed range while still cutting cost.

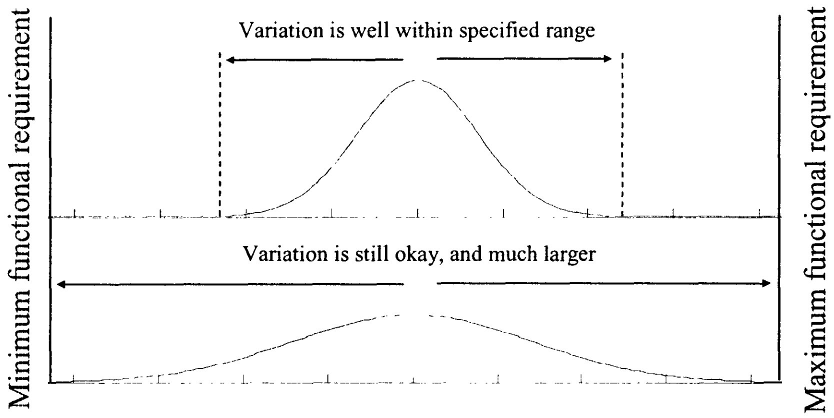


Figure 3.9: If a functional requirement lies too far within its specified range, it may be possible to cut cost by relaxing the tolerance on one or more of the input DPs

3.3.2.3 Cost-cutting opportunity: tighten the tolerance on a sensitive/highly coupled DP.

Coupling and variation are highly linked phenomena. If a design parameter affects multiple functional requirements, it can easily increase the net variation of the system. Components with such design parameters should not be employed unless it is unavoidable. However, it sometimes makes sense to design a system with shared components such as power supplies or thermal regulation subsystems if the larger physical integration constraints or the cost constraints in place warrant it. In these highly coupled components, it could actually be cheaper in terms of the big picture to tighten the tolerances on the component DPs so that the problem of reducing or increasing variation across all of the functional requirements does not have to depend on the coupled components.

3.3.2.4 Cost-cutting opportunity: Magnitude and variation can be traded off.

Because the magnitude and the variation of a DP can have completely independent effects on the cost of a system, it is possible to optimize cost in some cases by relaxing the tolerance of a DP and increasing the mean value to compensate, or vice versa. The information axiom does not specify how the performance of a system should look as long as it satisfies the functional requirements at all times. The three performance distributions shown in Figure 3.10 are examples of different distributions which all satisfy the same FR.

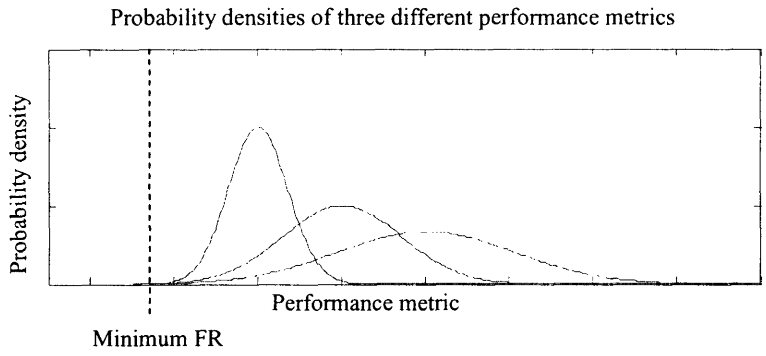


Figure 3.10: All three of the performance metrics plotted above satisfy the functional requirement shown (exceeding the minimum value at the left). As long as this is the only criterion governing success, the design parameters of the system can be varied in both magnitude and tolerance to optimize cost.

3.3.2.5 Stiffness of functional requirements is an aspect of design open to revision

If, for some reason, it is too expensive to eliminate variation in some component DP, always remember that the aspects of design other than the DPs of a system are still open to revision in the conceptual development phase. It is possible to design most systems to be forgiving of a great deal of DP variation by reducing the sensitivity of an FR to a particular DP. In practice this could be a matter of placing a particularly temperature-sensitive piece of electronics in a hot or cold environment directly next to the temperature sensor, so that the variation is minimized, compared to the variation in temperature which might be felt near the heater or at the extremes. In a hydraulic system which feeds extremely pressure-sensitive components, the effect of variation in pump output pressure could be mitigated through the addition of a larger pressurized reservoir and regulator. No amount of twiddling with design parameters can make up for a creative and robust design decision that reduces the effect of variation rather than worrying about the cause.

4 Conclusion

4.1 Summary

Design changes save money in a useful fashion when they impact the performance of a system in a predictable and acceptable way. A cost estimate produced by reading design details off of a blueprint and plugging them into statistical models is fundamentally still just a static number. Despite the ease with which such an estimate can be obtained, it provides no guarantee that a design change will accomplish what it sets out to without costly latent consequences. The tools discussed here are part of an attempt to remedy this situation by introducing functional thinking into cost estimation through axiomatic design. Through an emphasis on stating goals, basic modeling, and common-sense system representations such as the coupling matrix, axiomatic design empowers designers to make changes with a fuller understanding of the far-reaching implications which one small change can have. This functional foresight is the key to increasing the credibility of any recommendation, whether it is made for the purposes of cutting cost or improving reliability.

4.2 Further directions

Not all of the architectural cost implications are addressed through the method proposed here; in fact, there are several key sources of cost in a project which were not dealt with because they occur at a much later point in the design process. Chief among these are the constraining relationships which arise between components in the process of physically integrating a system. These relationships often have no functional component, but they can still create reliability and cost issues. For example, several independent components of a computer control system and a communications system might be housed together in the same enclosure. Once housed together, any changes to the dimensions or heat dissipation properties of any of these components would be linked to one another. Solutions to these physical integration coupling issues have been posed in the form of DP-DP interaction tables, which highlight the physical linkages between components using the same X and 0 notation used with coupling matrices. These DP-DP "matrices" would be then traversed in a fashion similar to the one outlined in section 3.3.1.1 for determining which components are affected by a physical change which would result in re-integration efforts. The other large cost source which it is not possible to model using the combined axiomatic design/parametric cost estimation approach is the overhead cost incurred by design changes, which become increasingly difficult as a project progresses. These costs can be initially approached by counting the number of DPs affected by a design change and using this number of impacted parameters to estimate the amount of engineering work which the change will incur; however, a more rigorous estimation method would probably resemble Smith and Eppinger's work transfer matrix approach (Smith & Eppinger, 1997), which assigns a weight to the amount of work incurred in each task due to a change in any other task. The thorough outline of a method like this has not yet been completed, but it shows promise.

5 Works Cited

Axiomatic Design: Advances and Applications
Suh, Nam P.
Oxford University Press
© 2001 by Oxford University Press, Inc.

Memorandum for Directors of Defense Agencies, Subject: Parametric Estimation
Office of the Under Secretary of Defense
August 23, 1999
http://www.ispa-cost.org/dodltr.htm

Space Mission Analysis and Design, 3rd Ed.
Wertz, James R. & Larson, Wiley J., Ed
Microcosm Press
© 1999 Microcosm, Inc. & W.J. Larson

Parametric Estimating Handbook, 2nd Ed.
International Society of Parametric Analysts
Spring 1999
http://www.ispa-cost.org/PEIWeb/newbook.htm

A Predictive Model of Sequential Iteration in Engineering Design
Smith, Robert P. & Eppinger, Steven D.
Management Science, Vol. 43, No. 8. (Aug., 1997), pp. 1104-1120.