Design Relations

by

E. Colburn, J.R. Rinderle

EDRC 24-33-90

Proceedings of the Second ASME
Design Theory and Methodology Conference
Chicago, Illinois
September, 1990

Design Relations

Eric R. Colburn
Sensor Frame Corporation
4516 Henry Street
Pittsburgh, PA. 15213

James R. Rinderle
Mechanical Engineering Department
Carnegie Mellon University
Pittsburgh, PA. 15213

Abstract
During preliminary design, designers make critical design decisions based on their experiential knowledge of the characteristics of the components that comprise a system. In particular designers understand the inherent relationship between the form a device will take and the behavior it will exhibit and use this knowledge to estimate characteristics of components and thus evaluate a system. Knowledge of these design relations are gained with experience, but even an experienced designer may not understand the design relations for an unfamiliar device. It is therefore useful to obtain this information automatically and provide it directly to the designer to supplement experience.

In this paper we discuss the nature of design relations and show that these relations are inherent in the physics, shaped by the constraints, and dependent upon the context of the design. We describe a method for automatically identifying these relations from a constraint-based model of a device which is based on solving a sequence of constrained optimization problems.

Introduction: A Design Scenario
Consider the design of a simple servo-positioning system that might be part of a business machine such as a personal computer printer or a copying machine. Two alternative configurations are shown in Figure 1. In Configuration A, the payload platform is positioned by a cable which is clamped to it. The cable passes over a stationary pulley and is driven by a motor which is mounted to the frame of the device. In Configuration B, the motor moves with the platform. The cable is wrapped around a capstan which is dead-ended at opposite ends of the positioning track.

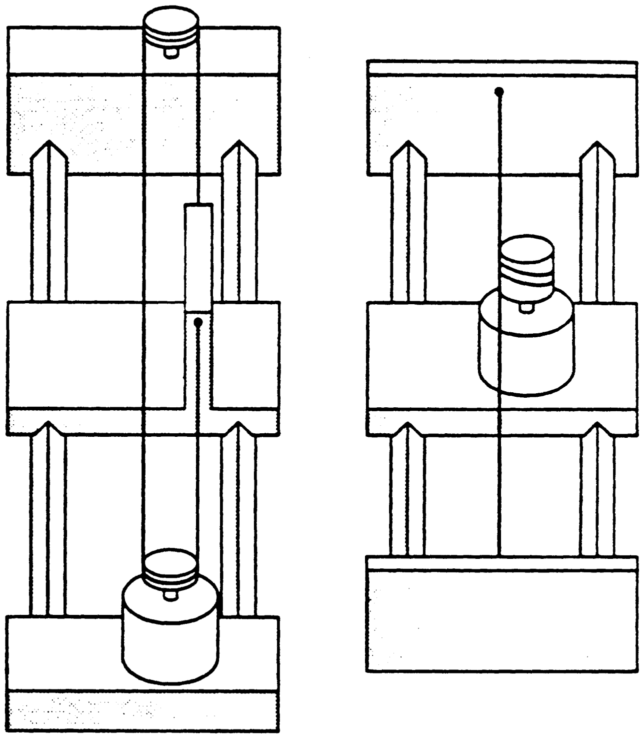

Figure 1: Two Servo-Positioner Design Configurations

Which is preferable? How does the designer choose between these two configurations? One possibility is to complete each of the two designs including the specification of the motor for each configuration. The designer may then select between the two substantially complete design alternatives on the basis of anticipated cost, performance, reliability or other criteria. Although selecting from substantially complete designs may be practical in this simple case, it does give rise to a duplicity of design, prototyping and evaluation efforts. Furthermore, in most instances completing all reasonable design configurations is not possible. Thus, in practice it is necessary for the designer to select among alternative configurations *prior* to design completion cognizant of the fact that selections made during this preliminary design phase are critical to the economic and functional viability of the product. How do designers select among alternatives prior to design detailing and how do they evaluate key characteristics of alternatives in sufficient detail so as to be able to make a *correct* selection?

Casually, we may discuss the characteristics of the two candidate configurations. In Configuration B, the entire mass of the motor is carried with the payload. It appears that a substantial performance penalty must result from this self-loading arrangement In Configuration A, the motor is stationary and therefore self-loading, as in Configuration B, is not present. On the other hand, Configuration A, requires a pulley and a longer cable which are not required in Configuration B. Furthermore, locating the motor at the end of the positioning track results in Configuration A being longer than Configuration B. The specific requirements of the product will determine the relative importance of these characteristics and will therefore, determine which of the two configurations is preferable, but in any case our estimate of these characteristics depends critically on the characteristics of the motors. How much will the motor weigh relative to the payload? How much space will the motor consume? The answer to these questions hinges fundamentally on the relationship between motor functionality, e.g. torque, and motor form, e.g. size and weight. An understanding of the relationship between motor form and motor function is necessary for the designer to intelligently select one of these two alternative configurations [Rinderle 87]. Relations such as these reflect the characteristics of a given device. They are abstractions of the physical reality of a device, and are useful because they help the designer understand the device and thereby integrate it into a system.

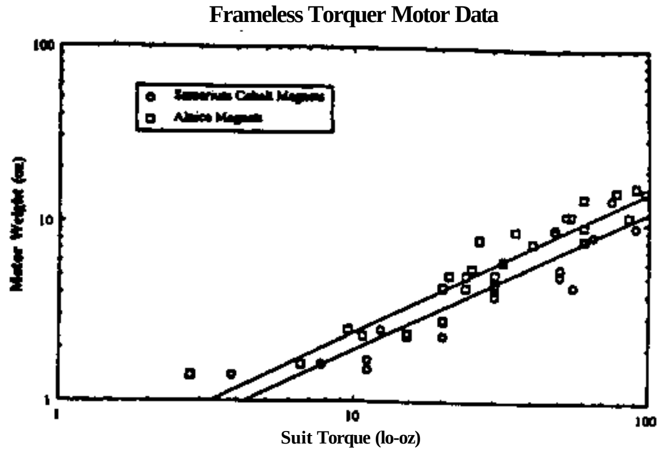

Figure 2: Motor Weight vs. Torque for Two Classes of D.C. Motors [Inland 85]

The Nature of Design Relations

A form-function relation, such as the motor torque-weight relationship, is an example of the broader class of design relations. Design relations are relations among any relevant design parameters. What is the nature of these relations? How are they obtained and used? These are the questions which we discuss in the remaining sections of this paper. We present a method for automatically identifying relevant relations, including a discussion of the representation employed. We then discuss how the resulting relations are dependent on and shaped by the constraints on the design of the device.

Design Relations as Abstractions

The servo-positioner designer is attempting to determine how overall device performance will depend on certain characteristics of the motor, particularly torque and weight. The designer docs not need or want to consider all characteristics of a motor but only those which he considers germane to particular device performance considerations. In this sense the designer is seeking a simplified view of a motor which includes only certain characteristics. We call this selective elimination of detail an *abstraction.* The designer, however, does not want the torque and weight of a specific motor but rather seeks a relationship between torque and weight for a family of motors. He wants an abstraction for a class of motors. Abstractions of this son can be obtained in many different ways. Most expert designers would rely on their experience in estimating the relative size and weight of a motor. A designer without adequate experience might rely on trends obtained from catalogue data. For example, Figure 2 is a plot of torque and weight for two classes of DC motors which might be considered for use in a positioner.

These abstractions make it possible for the designer to reason about the feasibility of the proposed configuration without considering the possibly irrelevant details of motor current, heat dissipation, mounting holes and the like. Designers come to know these relationships with experience, but even an experienced designer may not understand the subtle characteristics of an unfamiliar device. It is therefore useful to obtain this information automatically and provide it directly to the designer to supplement experience.

Designers use abstractions to reduce the number of concepts or variables which need to be considered simultaneously [Balling 85]. Furthermore, the observation that expert designers use more abstract representations than novices [Newsomc 88] indicates the importance of abstractions in design. Abstractions effectively reduce the number of variables which the designer must consider either by neglecting certain characteristics or constraints or by formulating an alternative view which combines effects into a simpler representation. The use of Reynolds Number instead of inertia! and viscous forces is an example of this type of abstraction and is discussed by [Watton-89]. The use of abstraction to control detail allows designers to focus their attention and frees them from having to deal with unnecessary complexities [Paz-Soldan 89].

Limitations of Experience Based Abstractions

It seems clear that the use of abstractions facilitates preliminary design, however, the validity of preliminary design assessments will depend on the validity of the abstractions employed. Empirical methods, however, are subject to error due to hasty generalizations and narrowness of experience. In addition, these relations are dependent on the state-of-the-art, therefore major changes in technology can render the knowledge of particular design relations obsolete. This is particularly an issue when experience is employed in an ad hoc fashion without the knowledge that the experience is directly applicable in a particular design context. Although it will be obvious to a designer that extensive experience with washing machine motors may not be applicable to the design of a servo-positioner, it is perhaps less obvious how experience with large servomotors will apply to smaller servomotors. Furthermore, any specific design context might have very special requirements which may alter the design relationships. For example, we would expect a significantly different relationship between torque and weight in the case where the axial space available for the motor was quite small and the radial space large, accommodating only a "pancake" type motor. It is clear therefore, that the context influences the appropriate abstractions for a class of device.

Descriptive Verses Prescriptive Design Knowledge

Design relations reflect the characteristics of a given device. They **are** abstractions of the physical reality of a device, and **are** useful because they help the designer understand the device **and** thereby integrate it into a system. They should be distinguished from design heuristics that capture knowledge of the design process. Procedural heuristics are suggestions or hints of how to design, they **are** a *prescription* for design as opposed to design relations which are a declarative *description* of a designed device.

Many design automation systems use procedural heuristics to emulate the way a human designer works. We **are** not attempting to automate design, but rather to automate the identification of design relations that will help human designers. This is in contrast with expert systems which use the design heuristics that humans have found to help the computer design. Thus we have reversed the roles of the computer and the human designer in this respect.

Obtaining Design Relations

Design relations reflect the physics of a component, the objectives of the component designer and the limitations on an acceptable design. The weight of a motor for example, would be determined by space limitations, the physical laws governing electro-mechanical devices, and the designers objective, e.g. to obtain a certain torque with minimum weight. By solving a *family* of component design problems we can ascertain a relationship among parameters: A family of motors designed for slightly different torque requirements makes it possible to determine a torque-weight relationship.

Although catalogue data such as that shown in Figure 2 can provide a basis for determining design relationships, the catalogue items do not reflect the special constraints or objectives of the current device level design problem. As an alternative to sole reliance on catalogue data, we explicitly represent the physical relationships for a component family. We then augment those constraints with a design criterion and problem specific constraints and solve the resulting component design problem. To obtain design relations, we then solve a series of these parametric design problems. Each different problem results in a different solution, and the series of solutions reveals the relationships between the parameters of the designs.

This method facilitates a compact representation of a family of components and an efficient means to identify many design relations within related design contexts. This is more practical than collecting a large database of designs or precompiled relations and is easier to update and modify. It also allows for reasoning about aspects of the device that are not listed in a catalogue and about designs that do not actually exist but could, if needed, be manufactured.

Representation

Just as the design of the servo-positioner can be based in pan on a declarative representation of motor relations, so to can the motor design be based on a declarative representation of the physics of motors and of the components which comprise motors.

We employ constraints among device parameters as the fundamental declarative representation of the device. The constraints arise from physical laws, spatial relationships and material limitations. Collectively these constraints delimit the space of acceptable designs, although both the parameterization and the choice of relevant constraints shape the design space. Satisfaction of these constraints, as has been discussed by [Gross 86], [Serrano 87] and [Simon 85] among others, is central to the design task.

In this model each parameter describes some characteristic of the form (such as a physical dimension or material density) or behavior (such as velocity, stress, or torque). The constraints relate the

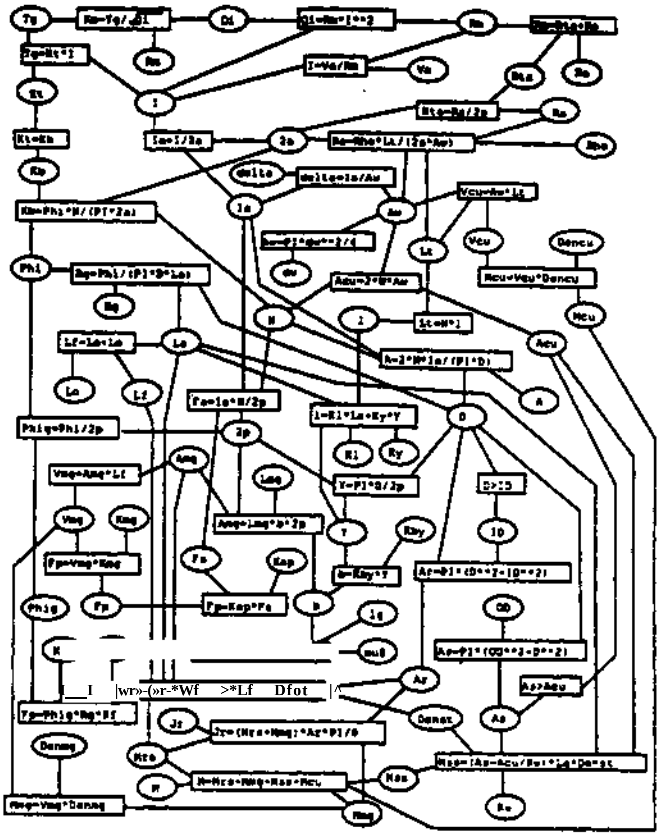

Figure 3: Constraint Network for a D.C. Electric Motor

parameters typically through equalities or inequalities.[1] Equality constraints are fixed relationships between parameters and may be the result of physical law (e.g., *f=ma)%* may be imposed as a requirement of the design (e.g., *voltage* * 12 *volts)* or may define a geometric relationship (e.g., $A =* nD^2/4$). Inequalities are often used to express physical limitations (e.g., *temperature* < *melting temperature),* imposed requirements (e.g., *torque* > 2.4 *ft-lbs)* or spatial relations (e.g., *OD* > *ID).* The compositional nature of the constraints allows the model to be easily expanded to an arbitrary level of description by the addition of parameters and constraints.

A collection of these constraints forms a network [Gross 86] or a *bipartite graph* [Serrano 87] with each node representing either a constraint or a parameter. In this graphical representation of the constraint model, each parameter node is linked to all of the constraints that it participates in, and each constraint is linked to all of its participating parameters. A constraint network for a brushless, unhoused *DC* motor under stall conditions is shown in Figure 3. Note that most of the inequality constraints and numerical limitations on the parameters are not shown in the figure for the sake of clarity.

The constraint network itself represents a prototypical device or class of devices and *satisfying* the network of constraints by assigning a value to each parameter such that none of the constraints are violated, results in an instance of the class. Thus we can view

[1]We are currently using only these but our approach does not preclude other constraints such as boolean, discrete, differential equations, etc.

the satisfaction of **a** constraint network as analogous to parametric **design.** **But** our goal in developing this representation is not to support parametric design, it is rather to aid the designer by providing **a way** to automatically identify relevant design relations.

Levels of Abstraction

We must represent the design **of a** motor at the level of the motor designer in order to support **the** identification of design relations that are useful to the servo designer. In general we must represent the device at **a** level lower than the design relations which we seek to identify. This is because the *levels of abstraction* employed by designers **are** relative. For example the servo designer will consider the motor **a** component that can be reasoned about as an abstraction while the motor designer will consider the motor a complex system consisting of components itself, such as bearings, wires and magnets each of which he will treat at some abstract level.

At each level designers will employ design relations that characterize the subsystems to facilitate reasoning about the device as **a** whole. Motor designers understand the heat/speed/load characteristics of bearings which help them specify the bearings and design the motor. Designers that use motors in their designs understand the torque/weight/power characteristics of motors that help them incorporate the motors in their systems.

Design Context

The design relations depend on the particular design problem. A network of constraints represents the nature of a class of devices, but not a particular design problem. The specific requirements and objectives of a problem constitute a *design context* which can be cast over the network to represent the given task. The context further constrains the space of the solutions to those that are acceptable for the particular situation and specifies the criteria necessary to identify the *best* design among these. Thus both a design and its corresponding set of design relations are directly related to the context under which it was created. For example one designer may be designing a motor for a given torque while trying to minimize weight, while another may be trying to maximize the efficiency while maintaining a fixed diameter. While both designers may be designing motors of the same class, the first will produce a different design than the second, and the relations between characteristics such as torque and weight or efficiency and diameter will be different for the two contexts.

The declarative nature of a constraint network allows these different contexts to be cast over it. This permits a variety of situations to be considered with a relatively compact representation. A large database of designs or precompiled relations need not be stored, but rather they can be generated as needed. In addition, designs that do not actually exist, but could be custom designed, can be represented, as can characteristics of the device that are not listed in a catalog or database. Thus the designer is able to reason about a wider variety of possibilities than may have otherwise been possible.

Method

To determine a design relation we solve a sequence of optimization problems, corresponding to a continuum of design contexts. This technique results in a series of optima which can be plotted to show the relationships between various parameters. The plot is similar to what [Siddall 82] calls *interaction curves.*

Our method consists of five basic steps:

1. Develop a network of constraints which represent the class of component.
2. Assert a design context by specifmg the requirements of the design and the objective.

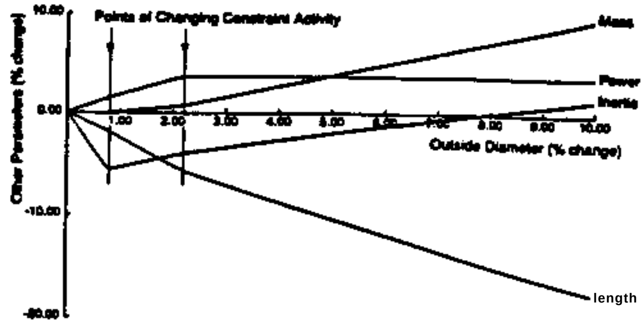

Figure 4: Design Relations for a D.C. Motor

3. Instantiate the constraint network based on catalogue data to obtain prototypical values for all parameters.
4. Optimize the prototype for the given specifications and objective.
5. Repeatedly perturb the design context by changing one constraint and optimize to obtain the trends in design variables.

The following example illustrates our approach.

Example: Brushless D.C. Motor

Consider the use of a frameless brushless DC motor in **a** servo-positioner as seen in Figure 1. The designer wishes to reduce the length of a standard motor to fit it in a particularly tight space. It is necessary to reduce the length as much as possible while maintaining the same torque and inside diameter and minimizing weight. The designer may be interested in increasing the outer diameter to effect this change, however, the designer will want to know how changing the diameter may affect some of the other aspects of the design.

Figure 4 shows some of the design relations obtained for this context from the constraint network shown in Figure 3. Each of the points marked on the curves represents a computer generated optimum of a prototype motor. Each curve then relates diameter of the motor to some other parameter. The relations are shown as relative changes in the outer diameter (measured along the abscissa) versus relative changes in the other parameters (measured along the ordinate). Thus the origin represents the standard motor (optimized for minimum mass), and the curves represent relative changes from that point.

The Relationship Between Constraints and Design Relations

It is apparent that the low-level constraints used to model the device prototype are related to the high-level design relations abstracted from the model. The connection between them can be demonstrated by considering the space of possible designs as defined by the collection of constraints.

The *design space* [Gross 86] can be measured in the dimensions of the design variables: the characteristics that a designer has direct control over, such as dimensions or materials. Each point in the design space represents a unique set of values of these variables, and thus a unique design. Furthermore, the region of feasible and acceptable designs is partitioned from the rest of the space by the inequality constraints. The space within these bounds contain designs which satisfy all the constraints.

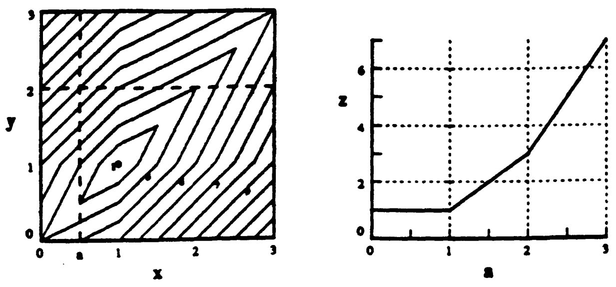

Figure 5: Effect of Inequality Constraints on Design Relations

Design can be viewed as moving through this space while attempting to improve a design until there is no way to do so without violating a constraint. The resulting point can then be considered an optimum because it is the design that best meets the objective and still satisfies all the constraints. Usually at the optimal point a number of inequality constraints are *active*; i.e., they have been pushed to their limits and hold as strict equalities. Each active inequality consumes one of the degrees of freedom, and often at the optimum all are consumed. This is called a *constrained optimum* in the optimization literature [Siddall 82, Balling 85] and is the common form of an engineering design solution.

Design relations are the result of the interaction of the defining constraints under the influence of an imposed context. In particular, the activity of the inequality constraints shape the relations as shown in Figure 5a in which contours of a function $z(x,y)$ are shown. We wish to find the minimum of z subject to two constraints, $y \leq 2$ and $x \geq a$. As the value of a is changed, the feasible region (depicted by shading in the figure) also changes. When $a \leq 1$, the minimum is not influenced by the constraints. As a is increased, the constraint forces the minimum away from the global minimum resulting in z becoming a function of a as shown in Figure 5b. When $a \geq 2$, both constraints are active and the change in constraint activity is reflected in a cusp in Figure 5b. Each point on the curve represents an optimal design for a different specification. Essentially we can see that the design relation is a trace of binding or limiting constraints. Thus design relations abstract the important characteristics of the design space relative to a given context and a variation in one of the specifications.

We can see this effect in the example illustrated in Figure 4. We note that relationships among design parameters change dramatically with changes in the activity of the constraints resulting in the various cusps in the plots. Of particular interest in this example is the relationship between outer diameter and the motor inertia. Initially increasing the diameter actually decreases the inertia, but the rotor will reach a limit on strength and further increases in the diameter will call for a more robust and therefore higher inertia rotor. Determining the nature of the active motor constraints in this way may help the servo-designer understand what is limiting the design and therefore help in reasoning about alternatives.

In general predicting the constraint activity is very difficult due to the complex, non-linear nature of the underlying equations and inequalities. The constraint activity depends on both the constraint network and the specific context of the design. For this reason it is not possible to determine *a priori* the active constraints, instead they must be determined for each context. Because they are difficult to predict and because they influence the design relations so strongly, the knowledge of the activity of the constraints is valuable to the designer.

The activity of the inequality constraints is important not only because of its influence on the design relations but because it can provide useful insight into the nature of the device. As the design solutions are generally constrained optima, the design is essentially limited by some aspect(s) of the technology or configuration. By providing the designer with a list of the active constraints he will be able to reason about what improvements may be made to the design, and, possibly just as important, what cannot be improved upon. For instance if the active constraints show that the design is limited by the strength of a member of the device it may be possible to change the material, but if the design is seen to be limited by imposed design specifications such as a given voltage available, then the designer can see that there is no room for improvement in that area.

Designers often characterize designs in this way, for instance a device may be called a "stress-limited" design or a "buckling-limited" design. This characterization of components and subsystems is useful when considering the design of the device because it faciltates reasoning about alternatives. Knowing that the component is limited in a specific way may tell the designer to discard the current configuration because the component can never perform as desired.

While we can see that constraints on a design and design relations are closely related, there are important differences, particularly operational differences. Constraints are used to model the low-level physics of a device. They relate "local" parameters directly through physical or geometric laws, for example $F=ma$, $v=iR$ or $OD=ID+2t$. They are a direct representation of the physics. Design relations, on the other hand, are used to express the high-level interaction of two characteristics of a device. They are useful because they relate two seemingly "distant" parameters, such as torque and diameter, in a simple way.

Conclusion

Empirical knowledge of device characteristics used by experienced designers can be seen to be abstractions of the underlying physics of a device and the objective of the device designer. These characteristic design relations are used by designers to evaluate tenative configurations based on estimations of the form and behavior of the components that constitute the device. Unfortunately, knowledge of these relations is obtained only with years of experience. Thus it is advantageous to automatically obtain relevant design relations as a means of supplementing a designer's experience. The method presented for doing this is based on the declarative representation of the constraints on the design and the use of optimization to produce design instances for a continuum of design contexts.

Acknowledgments

The authors are pleased to acknowledge the support of the Design Theory and Methodology Program of the National Science Foundation (NSF Grant DMC-84-51619) and the Engineering Design Research Center at Carnegie Mellon University (NSF Grant CDR-85-22616).

References

[Balling 851

Balling, R. J., Free, J. C, Parkinson, A. R., "Exploring Design Space: Optimization as Synthesizer of Design and Analysis," *Computers in Mechanical Engineering,* March 1985, pp. 28-36.

(Gross 86)

Gross, M. D., *Design as Exploring Constraints,* PhD dissertation, Department of Architecture, Massachusetts Institute of Technology. 1986.

[Inland 85]

Inland Motor, KoUmorgen Corporation, "Direct Drive DC Motor Catalogue".

[Newsome 88]

Newsome, S. L. and Spillcrs, W. R., "Tools for Expert Designers: Supporting Conceptual Design," in *Design Theory '88.* Newsome, S. L., Spillcrs, W. R., and Finger, S., ed.. Springer-Veriag. New York, 1988.

[Paz-Soldan89]

Paz-Soldan, J.P. and Rmdcrlc, J.R., "The Alternate Use of Abstraction and Rcfincmcni in Conceptual Mechanical Design," *AS ME Paper No. 89-WAIDE-8,*1989.

[Rindcrle 87]

Rinderie, J. R., "Function and Form Relationships: A Basts for Preliminary Design," *Proceedings from the SSF Workshop on the Design Process,* Waldron, M. B.. ed., Ohio State University. Oakland, CA, February 8-10 1987, pp. 295-312.

[Serrano 87]

Serrano, D., *Constraint Management in Conceptual Design,* PhD dissertation. Department of Mechanical Engineering , Massachusetts Institute of Technology. 1987.

[Siddall 82]

Siddall, J. N., *Optimal Engineering Design,* Marcel Dckker. New York. NY, 1982.

(Simon 85)

Simon, H. A., *The Sciences of the Artificial.* MIT Press. Cambridge. MA. 1985.