WORKING PAPER

ALFRED P. SLOAN SCHOOL OF MANAGEMENT

Testing an Engineering Design Iteration Model in an Experimental Setting

Robert P. Smith
Steven D. Eppinger
Amarnath Gopal

February 1992
WP #3386-92-MS

MASSACHUSETTS
INSTITUTE OF TECHNOLOGY
50 MEMORIAL DRIVE
CAMBRIDGE, MASSACHUSETTS 02139

Testing an Engineering Design Iteration Model in an Experimental Setting

Robert P. Smith
Steven D. Eppinger
Amarnath Gopal

February 1992
WP #3386-92-MS

Acknowledgement

This research was funded by the Leaders for Manufacturing Program, a partnership involving eleven major U.S. manufacturing firms and M.I.T.'s schools of engineering and management.

Keywords: design methods and models, design management

Send correspondence to:
Prof. Steven D. Eppinger
M.I.T. Sloan School of Management
30 Wadsworth Street, E53-347
Cambridge, Mass. 02139

Abstract
In this paper, we compare two alternative design strategies for the Delta Design Game, an engineering design exercise. We first analyze these strategies using the Work Transformation Matrix, a design iteration model which shows that one of the strategies is expected to display a faster solution time. We then demonstrate experimentally the difference in development time by observing eight design teams working on the problem using the two strategies. We found that the "decoupling strategy" suggested by the model reduced solution time while maintaining quality of the technical solutions.

1. Introduction

Design performance is an important factor in determining the success of a manufacturing firm. The amount of time that it takes the firm to develop a product is an important factor in determining the success of the design [Clark and Fujimoto 1991].

Our study of the design process has led to the development of formal mathematical models of the design process which can estimate the amount of time that it takes to design a technical product [Eppinger *et al.* 1990, Eppinger *et al.* 1992, Smith and Eppinger 1991a, Smith and Eppinger 1991b]. We have applied one of the models (the Work Transformation Model, described in the next section) to some industrial design environments [Smith and Eppinger 1991b], which have shown reasonable correlation with our model. While we are encouraged by the analytical results of these design iteration models, we have found, however, that actual design environments are difficult to observe directly. One must rely on the retrospective description of the design process from the individuals involved.

We have designed the research experiment presented here in order to verify the ability of the Work Transformation Model (WTM) to predict important differences in the performance of alternative design strategies. The experimental design environment is small enough so that we can observe and control it directly. We will show that: (1) the model is able to suggest a successful decoupling strategy which is not obvious to the design engineers; and (2) the model is able to identify critical technical issues which drive iteration time.

Experiments have been used in other settings to test hypotheses about what affects the ability of designers to accomplish their task [Jakiela and Orlikowski

1990, Papalambros 1988]. Experimental settings increase the ability of the researcher to control the design environment, and to gather many data rapidly.

2. Theory of Design Models

2.1. Design Structure Matrices

The model of the design iteration process is based on the Design Structure Matrix (DSM), suggested by Steward [1981] as a useful tool for identifying the interdependent information flow inherent in engineering design.

The Design Structure Matrix (DSM) serves as the basis for our formal analysis and will be briefly reviewed in this section. (For a more detailed overview of the DSM method the reader is referred to Steward [1981] and Eppinger *et al.* [1990].) The work herein describes the analytical method, and demonstrates correspondence between predictions of the model and experimental reality.

The philosophy of the DSM method is that the design project is divided into individual tasks, and the relationships among these tasks can be analyzed to identify the underlying structure of the project. It has been suggested that studying the relationships between individual design tasks can improve the overall design process, and is a powerful way to analyze alternative design strategies [von Hippel 1990]. Earlier work developed a modeling formalism which shows how different aspects of a design problem are related [Alexander 1964]. The DSM method is a more formal and complete model than Alexander's.

In the DSM method, specified tasks are arranged in a square matrix where each row and its corresponding column are identified with one of the tasks. Along each row, the marks indicate from which other tasks the given task requires input. Reading down each column indicates which other tasks receive its output. Diagonal elements do not convey any meaning at this point, since a task cannot depend upon its own completion. For example, in Figure 1 (based on a simplified view of camera body design), task **C** requires input from tasks **B**, **D**, **E** and **F**, task **B** requires input only from task **A**, and task **A** needs no input to begin.

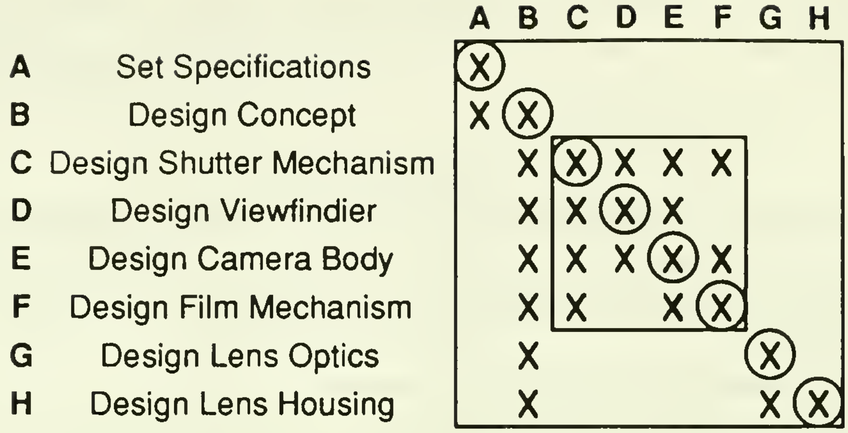


Figure 1. Sample Design Structure Matrix

The DSM can be used to identify orderings of tasks and to identify difficult aspects of the design process. Some or all of the elements of the matrix can be made sub-diagonal (such as those corresponding to tasks **A**, **B**, **G**, and **H** in Figure 1) by reordering the tasks of the matrix using a partitioning algorithm [Steward 1981, Gebala and Eppinger 1991]. An entirely sub-diagonal matrix indicates that there exists a sequence where all tasks can be completed with all input information available. Such a sequence may contain both tasks which must be done in series, or tasks which may be done in parallel. The information in a sub-diagonal design matrix is then similar to that expressed in a CPM (Critical Path Method) chart.

More typically, due to the complexity in engineering design, the matrix cannot be reordered to have all matrix elements sub-diagonal (such as tasks **C-F** in Figure 1.) In these cases there is a cyclic flow of information in the design process and standard CPM techniques are not applicable because of the presence of such cycles. Likewise, a sequential progression of the design tasks is not possible. Tasks where neither a purely sequential nor a parallel ordering is feasible are coupled in such a way that some alternative process for resolving the design interactions (such as iteration or negotiation) must be used. The sub-matrix in Figure 1 depicts a design problem defined such that the tasks are sufficiently complex and interrelated so that iteration will be necessary to complete the tasks.

If we include task durations in the DSM, we can use this description to estimate the total duration of the project. Series tasks can be evaluated by summing their individual times, and parallel tasks can be evaluated by finding the maximum of those task times. For the project characterized by the DSM in Figure 1, if the task time are a, b, c, ... , h, the time of the camera design project would be

$$a + b + \max\{ f(c,d,e,f) , g+h \}$$

where $f(\cdot)$ is a function, undefined as yet, corresponding to the development time for the coupled block. In the next section we propose a model which will enable the calculation of development time for a coupled block.

2.2. Work Transformation Model Development

The model presented in this paper illustrates how iteration time can be evaluated for such a coupled block of tasks, and shows that the critical features controlling the iteration can be identified. Each critical feature is a group of parameters of the design solution which are strongly dependent on each other; they may require many iterations to converge, as a set, to conform to design constraints.

We use a modified version of a fully coupled Design Structure Matrix which we call the *Work Transformation Matrix* (WTM). The diagonal elements in the WTM represent the time that it takes to complete each task during the first iteration stage. (See Figure 2.) The off-diagonal elements represent strength-of-dependence measures (defined in next section). It is assumed that there will be multiple iteration stages, and that the time for each stage is a function of the amount of time spent working in the previous stage. We wish to find the sum of the times of all stages.

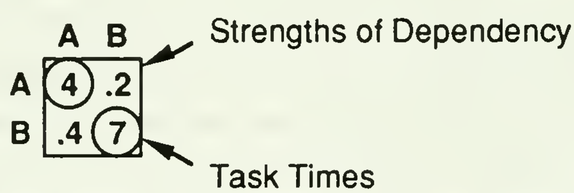


Figure 2. Work Transformation Matrix

We describe the model and its application in the following sections. In the first section we briefly discuss the assumptions underlying the model and how we interpret the results of the model. Following the description of the model, we illustrate the analytical process using a simple example.

2.3. Work Transformation Model Assumptions

The assumptions in this model are:

- All tasks are done in every stage – fully parallel iteration
- Rework created based on a linear rule – as a % of task
- The parameters in matrix describing work transformation behavior do not vary with time

These assumptions allow us to use a linear algebraic analytical method on the WTM.

To develop the model, we first introduce the concept of the work vector u_t. This is an n dimensional vector, where n is the number of design tasks to be completed. Each element of the work vector contains the fraction of work to be done on each task after t stages. The initial work vector u_0 is a vector of ones, which indicates that all of the work remains to be completed on every task.

The total work vector U is the sum of the work vectors for each stage. This vector contains the total number of times that the tasks must be completed during the iteration process.

We define a *design mode* as a group of design tasks which are very closely related, and working on any one of them creates significant work, directly or indirectly, for each of the other tasks within the mode. The design modes can be identified using the eigenvalues and eigenvectors of matrix A. (For more details on why the eigenvalues and eigenvectors are relevant to the calculation of vector U, see Appendix A.)

2.4. A Simple Example

As an illustration of how the modal analysis is applied, let us consider the following 4x4 Work Transformation Matrix. This is a quantitative version of the coupled block (tasks **C-F**) in the camera design matrix originally shown in Figure 1. The tasks in this matrix are, in order: Design Shutter Mechanism, Design Viewfinder, Design Camera Body, and Design Film Mechanism. The numbers can be interpreted as follows: if the shutter is completely redesigned, then 10% of the viewfinder design work must be redone (and so forth).

$$A = \begin{bmatrix} 0 & 0.1 & 0.2 & 0.6 \\ 0.1 & 0 & 0.3 & 0 \\ 0.1 & 0.4 & 0 & 0.1 \\ 0.6 & 0 & 0.2 & 0 \end{bmatrix}$$

The eigenvalue (Λ) and eigenvector (S) matrices are:

$$\Lambda = \begin{bmatrix} 0.71 & & & \\ & -0.61 & & \\ & & -0.34 & \\ & & & 0.25 \end{bmatrix} \qquad S = \begin{bmatrix} \mathbf{1.00} & \mathbf{-0.97} & -0.32 & -0.64 \\ 0.34 & 0.24 & \mathbf{-0.78} & \mathbf{0.96} \\ 0.47 & -0.16 & \mathbf{1.00} & \mathbf{1.00} \\ \mathbf{0.97} & \mathbf{1.00} & -0.02 & -0.76 \end{bmatrix}$$

The four eigenvectors are the columns in S. They are (arbitrarily) scaled to have the largest entry in each row be 1.00. By inspection of the eigenvectors, we learn that the most slowly converging design modes (the eigenvectors corresponding to the two largest magnitude eigenvalues) involve primarily the first and last tasks (Shutter Design and Film Mechanism Design, the highlighted values), and the two faster design modes involve mainly the other two tasks (Camera Body Design and Viewfinder Design). When we compute the first few remaining work vectors, we find that they support the above interpretation.

$$u_0 = \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \end{bmatrix} \quad u_1 = \begin{bmatrix} 0.9 \\ 0.4 \\ 0.6 \\ 0.8 \end{bmatrix} \quad u_2 = \begin{bmatrix} 0.64 \\ 0.27 \\ 0.33 \\ 0.66 \end{bmatrix} \quad u_3 = \begin{bmatrix} 0.49 \\ 0.16 \\ 0.24 \\ 0.45 \end{bmatrix} \quad u_4 = \begin{bmatrix} 0.33 \\ 0.12 \\ 0.16 \\ 0.34 \end{bmatrix} \quad u_5 = \begin{bmatrix} 0.25 \\ 0.08 \\ 0.12 \\ 0.23 \end{bmatrix}$$

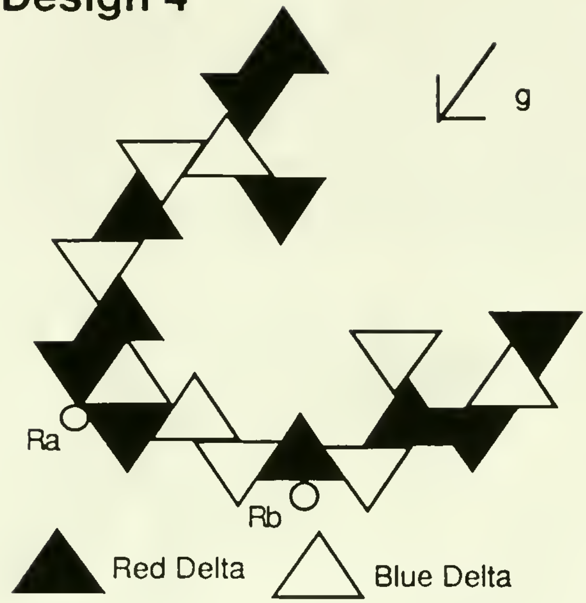
The work remaining on the middle two tasks is converging to zero more rapidly than the work on the other two tasks. We see that the dominant mode describes the shape of convergence of the remaining work vectors.

3. Delta Design Game

The Delta Design Game is an exercise originally used to demonstrate to engineering students that design is a process of negotiation among several conflicting disciplines and requirements [Bucciarelli 1990]. We have adopted the game for our purposes of demonstrating how two different design strategies for the same design problem can be judged using the Work Transformation Matrix.

In the Delta Design Game, four designers are responsible for designing a residential structure in a two dimensional world. (See Figure 3 for an example design.) (More details about the game can be found in Appendix B.) The building materials are triangular elements (deltas) which must be combined subject to a set of design specifications (thermal, structural, aesthetic and fiscal). Deltas come in both red (heat source) and blue (heat sink) varieties. Each of the participants is assigned a role in the design process, and associated with each role is some specific technical knowledge about the specifications. The roles are Thermal Engineer, Structural Engineer, Architect, and Project Manager.

Design 4



1		Design 4	
2			
3	Time(min)	55	
4	Total Quality	10.12	
5			
6	Parameters:		Quality pts:
7			
8	Cost	497.89	1.15
9	Area(Deltas)	26	1.05
10	% blueness	45	1.225
11	Ra and Rb	9 and 11	
12	Load		1.4
13	Overlap safety		1
14	Avg temp	58.6	1
15	Max temp	0'k	1
16	Min temp	0'k	1
17			
18	Aesthetics		1.3

Figure 3. Example Design

The game takes about two hours to complete, including thirty minutes to train the participants in their various roles. During the training the participants do not interact, so that their areas of expertise remain distinct.

3.1. Alternative Design Strategies

The standard design strategy, as described in the original instructions to the game, is as follows. The architect suggests a structure by arranging the deltas in a proposed design layout. The three other participants analyze the aspects of structure under their domain, and suggest design improvements. A new layout is proposed and the analysis is repeated. This iteration process continues until the design takes its final form (meets the target specifications). This process is shown in Figure 4.

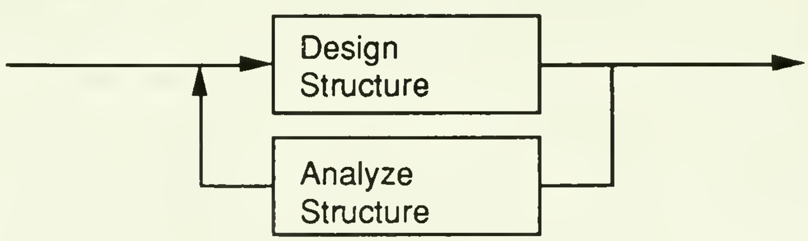


Figure 4. Standard Design Strategy

The standard strategy can be represented by a Work Transformation Matrix as follows. (See Figure 5.) The first five tasks in the matrix are the design tasks (where the participants lay out a suggested design). The other ten tasks are the analysis tasks, where the participants judge the design against the given criteria. There are no direct information flows from one design task to another, nor is there a direct dependency from one analysis task to another. Nevertheless, the matrix is fully coupled. The strengths of the dependencies were determined by the authors based on their experience with the design game, both as participants and observers.

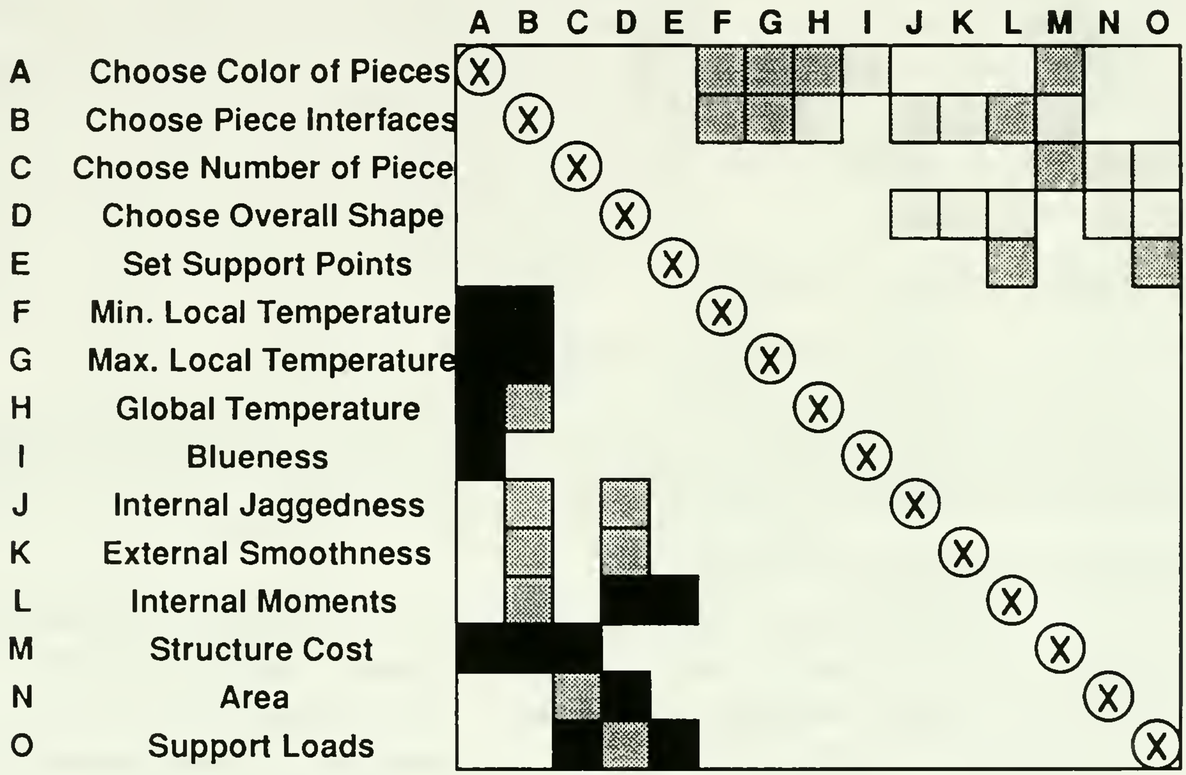


Figure 5. Matrix for Standard Strategy

Analyzing the eigenvalues and eigenvectors of the above matrix identifies the issues driving the design iteration. The eigenvectors corresponding to the largest eigenvalue (primary design mode) is primarily composed of tasks **A**, **B**, **F**, **G**, **H**, and **M**. (See Figure 6.) These tasks are associated with the thermal design problem and the cost of construction. The eigenvector corresponding to the second largest eigenvalue (secondary design mode) is primarily composed of tasks **E**, **L**, and **O**. These tasks are associated with the structural design problem (loads and moments.) We see that these two design modes are somewhat independent (i.e. they have different tasks which are heavily weights.)

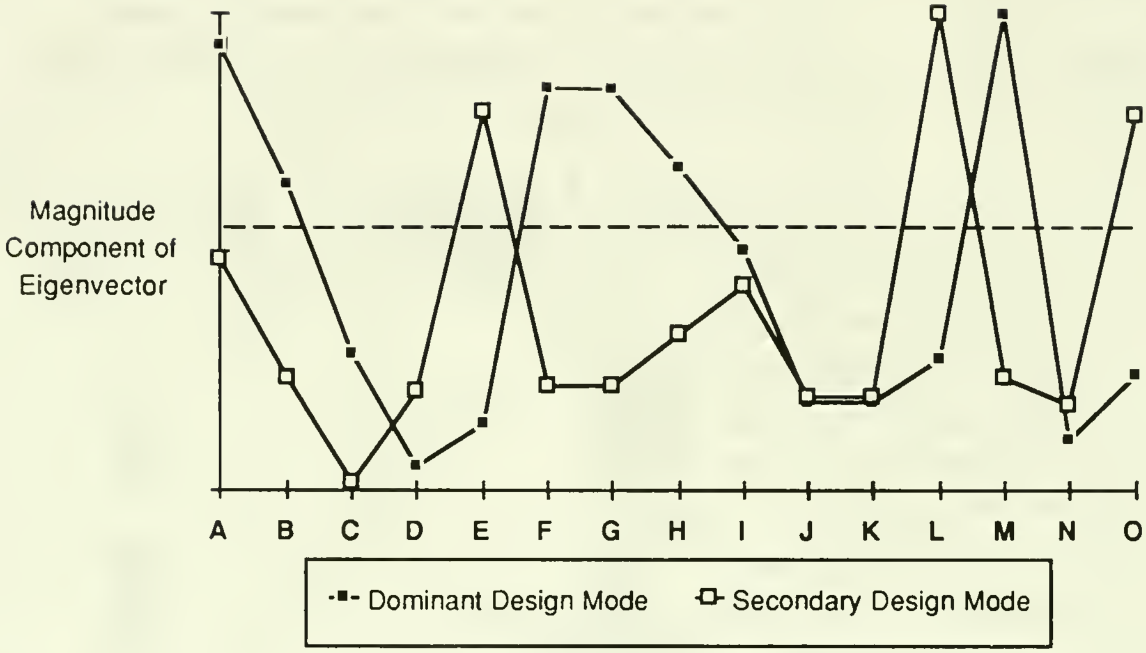


Figure 6. Dominant Eigenvectors of Standard Strategy

One method of working on a complex problem where there is some independence is to split the problem into smaller pieces, and to work on the pieces separately [Suh 1990]. We identified a design strategy which would allow the designers to separate the two design modes. The cost and thermal design criteria scale reasonably well from a small portion of the structure up to the entire building.

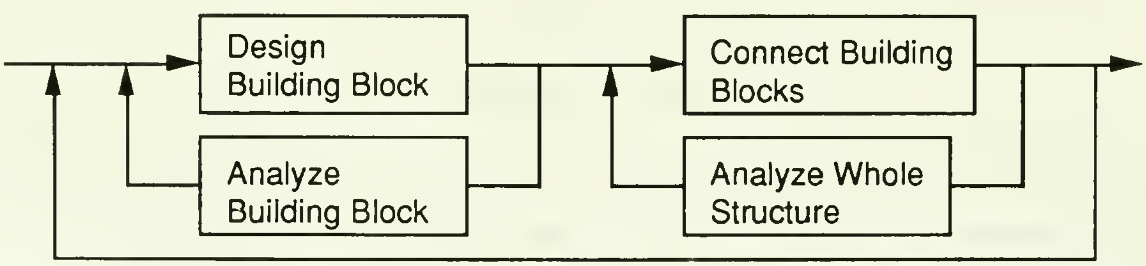


Figure 7. Building Block Strategy

This idea can serve as the basis for a design strategy. (See Figure 7.) During the first phase of the design process, the design team would try to find a building block (group of 2-6 deltas) which meets the local thermal criteria, and seems likely to produce a lower cost structure. This building block would then be replicated, and the blocks joined into an overall structure. The complete structure would then be checked against the structural criteria. If the building fails to meet any of the criteria, either the building blocks would be joined in a new configuration, or the building block itself would have to be redesigned. This process would be repeated until the structure was complete.

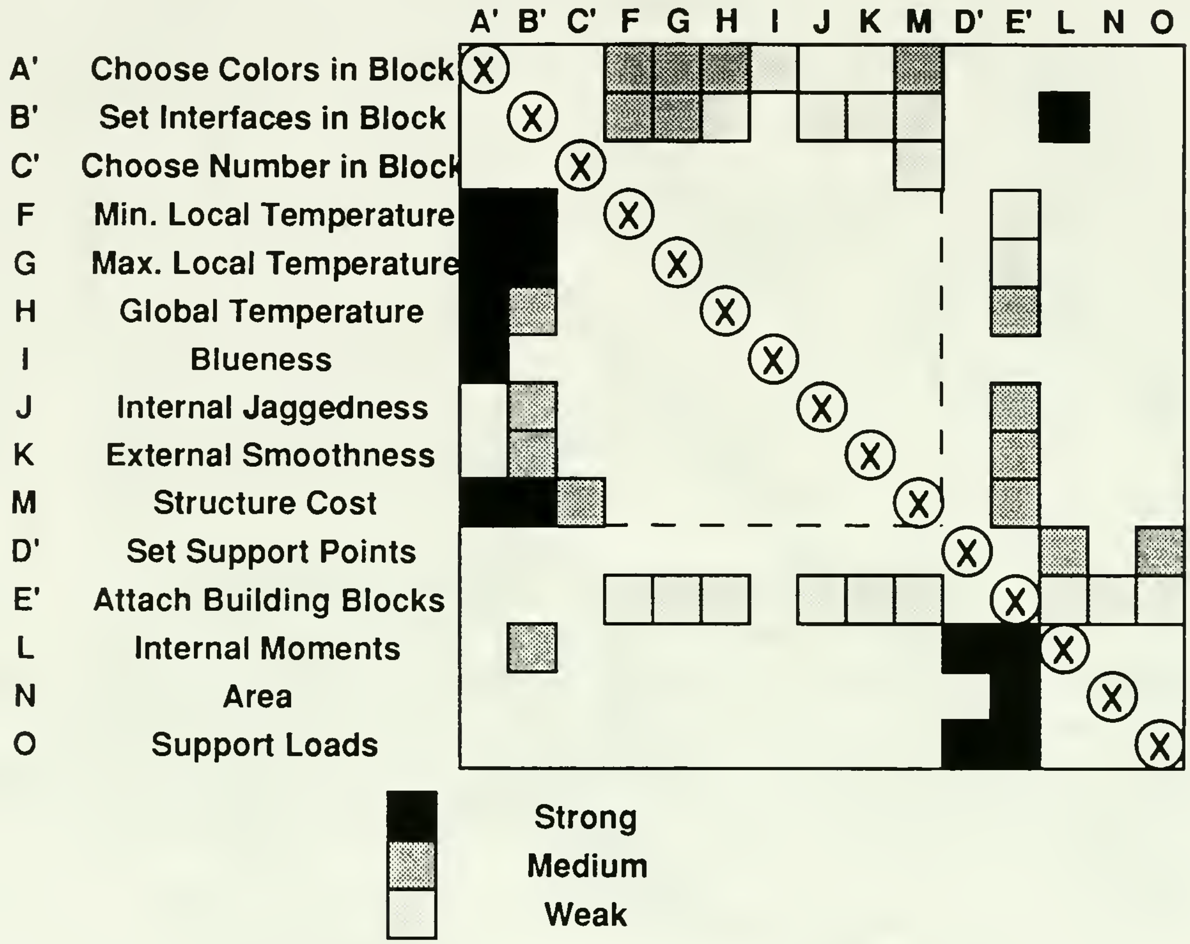


Figure 8. Matrix for Building Block Strategy

The building block strategy was also represented as a Work Transformation Matrix. (See Figure 8.) The building block matrix is similar to the matrix for the original strategy, although a few of the tasks have to be redefined and resequenced. The matrix is still fully coupled.

Looking at the dominant eigenvectors of this matrix, we see that the two primary design modes for this matrix are independent, as we expected when we selected this design strategy. (See Figure 9.)

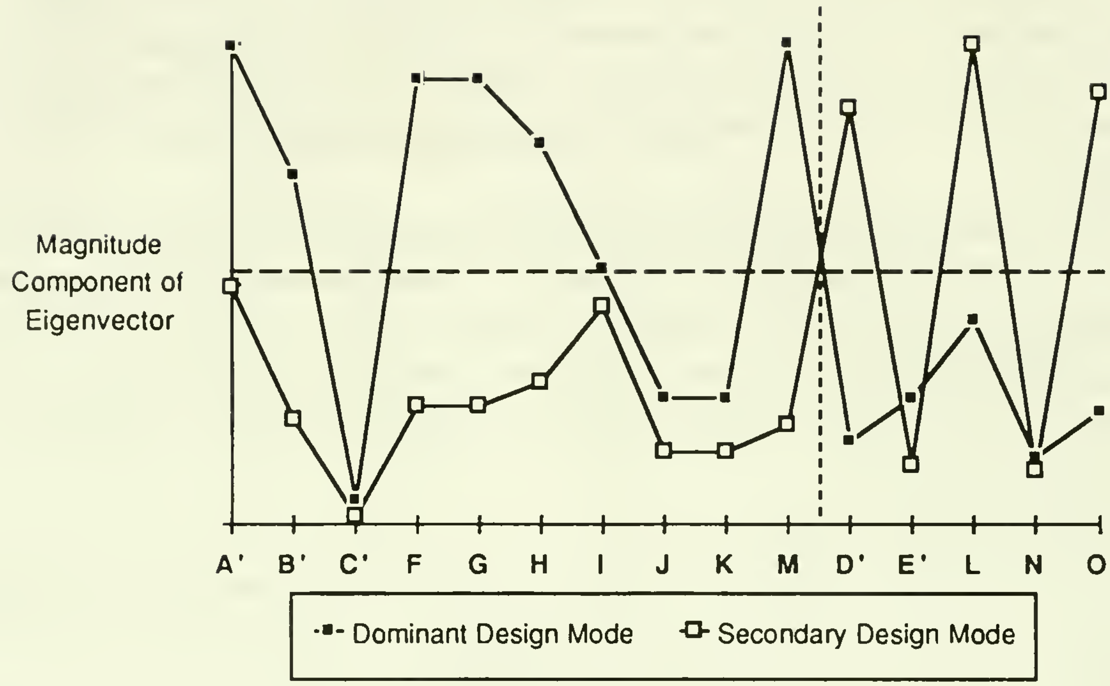


Figure 9. Dominant Eigenvectors for Building Block Strategy

Since the eigenvalues are most closely associated with how many iterations it takes to find the design solution, these measures are not able to discern which is the superior design strategy, in terms of fewer iterations. We have not attempted to assign the tasks with times. We believed that the building block strategy would dominate the original strategy because it takes a shorter amount of time per iteration, not because it will take fewer iterations. Comparing the tasks, we would expect it to take less time to choose interfaces which meet the thermal constraints when there are 2-6 deltas than for the complete matrix.

3.2. Experimental Method

In implementing the experiment, we wish to test to see if there is a significant difference in design time between groups who use the two strategies.

We formed eight groups of four undergraduate engineering students. Each member of each group was instructed in the overall nature of the design problem, their area of expertise, and the design strategy they were to employ.

For each group, we observed them performing the design task and we recorded the total time taken until the group produced its final design. We then recorded the final design in order to calculate a quality score. This was needed to see if there were any differences in the quality of the produced designs.

The quality score codifies our attempt to unify all of the various design constraints and criteria into one measure. The experimental groups did not use the quality score to evaluate the designs. They designed to the original design criteria. The quality score evaluation was done subsequently, for our own purposes.

The quality score penalized groups for those criteria which they failed to fulfill, while rewarding groups which exceeded design guidelines. For several of the constraints, there were two levels of constraint, such as a desired cost goal, with a 10% allowance, if necessary. On these types of constraint the penalty was not large, unless the group failed to meet the relaxed constraint, for which they were penalized heavily.

3.3. Experimental Results

The graph below (Figure 10) shows the relative times and quality scores for the eight design groups. We observe that the Building Block design teams took significantly less time than the Baseline design teams, while there is no significant difference in quality between the approaches. The significance of the differences between the mean times was tested using a two-sample t-test [DeGroot 1986]. The difference between the means of the times for the standard strategy (84.25) and the building block strategy (55.25) is statistically significant at a 0.005 level. The difference between the mean of the quality score for the standard strategy (9.30) and the building block strategy (9.64) is not significant, even at a 0.20 level of significance.

(The data point which is slower and of lower quality than the other points is explained as follows. The groups were told it would take 60-90 minutes to complete the design. They knew after 105 minutes that they were taking more time than allotted, although their design was improving. It is our hypothesis that this group's quality score would have come more in line with the other groups', at the expense of still more time.)

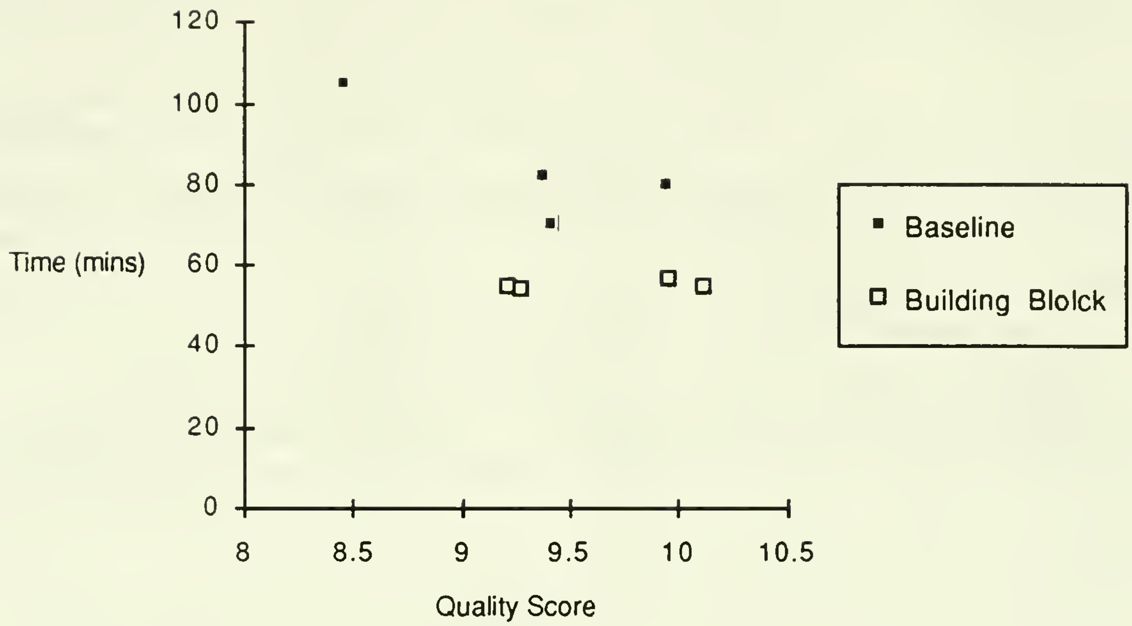


Figure 10. Time vs. Quality

There are not any other significant differences in the design results between the two design approaches. Complete experimental data are presented in [Gopal 1992], and are available upon request.

4. Discussion

The building block approach does not fully decouple the design problem; there is still information transfer required between the two design sub-problems. The most important feature of choosing the building block (first phase) which affects the structural design (second phase) is the setting of sufficient overlap between the deltas, so that the internal moment design criterion can be more easily fulfilled.

By suggesting a building block design strategy, we may have given more information to the design groups, explicitly or implicitly. What is important is that the two approaches are significantly different, and that the differences described in the matrix account for a significant portion of the difference.

We are not able to suggest design strategies directly from observation of the the mathematics of the matrix. Rather, we require direct knowledge of the design environment. This is not necessarily bad; we would not necessarily expect design structure analysis to be able to improve the design without domain-specific knowledge. The observation of the independence of the main design modes can, however, suggest to the designer where to look for strategies which will separate the modes into smaller problems.

We are led to wonder under what conditions decoupling is appropriate. In the delta design problem we were able to identify a decoupling strategy because of the independence of the design modes. We hypothesize that the independence of the primary design modes indicates that a decoupling strategy is technologically feasible.

Not all decoupling strategies will necessarily improve the design process. If the problem is divided across one or more key technical issues, then coordination between the two parts of the design process becomes an important and difficult task. It may be necessary to redefine some of the design tasks to facilitate decoupling. Also, it is often useful to look for opportunities to use scaling in order to decouple a design problem.

Industrial design environments we have studied (brake system design [Smith and Eppinger 1991b] and electronics module design [paper forthcoming]) have had design matrices which exhibited some degree of independence among the main design modes. The organizations did not exploit this independence specifically in order to solve their particular design problem. Both of these studies were conducted at firms which have extensive experience with their technical problems, and they feel that they have a good grasp of the technical issues which control their design problem. The goal of this modeling is to help identify the important features which control design iteration, which can help improve the management of design projects.

5. Conclusion

We have been developing models of the design process in order to increase our understanding of design and to provide managers with tools that they can use to improve their control of such projects. This paper tests the validity of the Work Transformation Model by comparing the predictions of the model with observation of design in an experimental setting. The model is able to predict the relative amount of time for two strategies to solve a design problem. The experimental data support the superiority of the design strategy. The model suggests that the times differ because of a greater amount of independence between design subproblems in the superior strategy. The model can also help to identify where a design strategy exhibiting such independence is possible or likely. Our future work will attempt to make the identification of superior strategies more structured. We also hope to test the ability of the design model to predict the success of a design strategy in an industrial design setting.

References

Alexander, Christopher, *Notes on the Synthesis of Form*, Harvard University Press, Cambridge, 1964.

Bucciarelli, L. L. and G. Goldschmidt, "Delta – A Design Exercise," Science, Technology and Society Program, M.I.T., 1989.

Bucciarelli, Louis L., "Ethnographic Study and Simulation of the Engineering Design Process," *Proceedings of the International Ergonomics Association: Human Factors in Design for Manufacturability and Process Planning*, pp. 61-66, August 1990.

Clark, Kim B., and Takahiro Fujimoto, *Product Development Performance: Strategy, Organization, and Management in the World Auto Industry*, Harvard Business School Press, Boston, 1991.

DeGroot, Morris H., *Probability and Statistics*, Addison Wesley, Reading, 1986.

Eppinger, Steven D., Daniel E. Whitney and David A. Gebala, "Organizing the Tasks in Complex Design Projects: Development of Tools to Represent Design Procedures," NSF Design and Manufacturing Systems Conference, Atlanta, 1992.

Eppinger, Steven D., Daniel E. Whitney, Robert P. Smith and David A. Gebala, "Organizing the Tasks in Complex Design Projects," Second International ASME Design Theory and Methodology Conference, Chicago, 1990.

Gebala, David A., and Steven D. Eppinger, "Methods for Analyzing Design Procedures," Third International ASME Design Theory and Methodology Conference, Miami, 1991.

Gopal, Amarnath, "Using Design Structure Matrices to Suggest a Strategy for the Delta Design Exercise," Bachelor's Thesis, M.I.T. Department of Mechanical Engineering, February 1992.

Jakiela, Mark J., and Wanda J. Orlikowski, "Back to the Drawing Board?: Computer-mediated Communication Tools for Engineers," Second International ASME Conference on Design Theory and Methodology, Chicago, 1990.

Ogata, Katsuhiko, *State Space Analysis of Control Systems*, Prentice Hall, Englewood Cliffs, N.J., 1967.

Papalambros, P., "Interdisciplinary Experiments in Design Research and Education," International Conference of Engineering Design, Budapest, 1988.

Smith, Robert P., and Steven D. Eppinger, "A Model for Estimating Development Time of a Sequential Engineering Design Process," Sloan School of Management Working Paper 3160-90-MS, 1991a.

Smith, Robert P., and Steven D. Eppinger, "Identifying Controlling Features of Engineering Design Iteration," Sloan School of Management Working Paper 3348-91-MS, 1991b.

Steward, Donald V., "The Design Structure System: A Method for Managing the Design of Complex Systems," *IEEE Transactions on Engineering Management*, Vol. EM-28, No. 3, pp. 71-74, 1981.

Suh, Nam P., *The Principles of Design*, Oxford University Press, New York, 1990.

von Hippel, Eric, "Task Partitioning: An Innovation Process Variable," *Research Policy*, Vol. 19, pp. 407-418, 1990.

Appendix A: Interpreting the Eigenstructure

This appendix contains some details on how the total work vector U is calculated, including discussion on why the eigenvalues and eigenvectors of A are useful in analyzing the iteration process.

During each iteration stage all remaining work is attempted on all of the design tasks. However, work on a task will cause some rework to be created for all other tasks which are dependent on that task for information. We determine which tasks those are from the design structure matrix. Every iteration stage produces a change in the work vector according to

$$u_{t+1} = Au_t$$

where each of the entries a_{ij} in A implies that doing one unit of work on design task j creates a_{ij} units of rework for design task i. The matrix A is then the work transformation matrix where the off-diagonal elements are given this interpretation and the diagonal elements are set to zero. The work vector u_t can be also be expressed by

$$u_t = A^t u_0$$

The sum of each of the work vectors is the total work vector U, the total number of times that each of the tasks is attempted during the total of T iteration stages of design process.

$$U = \sum_{t=0}^{T} u_t$$

$$U = \sum_{t=0}^{T} A^t u_0$$

The model output U is therefore in nominal units of iteration for each task. (If element i in vector U is 1.6, then the design organization will have done 60% rework on task i in subsequent stages.) We can scale U by the task durations to obtain units of task times. If W is a matrix which contains the task times along its diagonal, then WU is a vector which contains the amount of time (in engineer-hours) that each task will require during the first T iteration stages.

If A has linearly independent eigenvectors (the eigenvector matrix S is invertible) then we can decompose A into

$$A = S\Lambda S^{-1}$$

where Λ is a diagonal matrix of the eigenvalues of A, and S is the corresponding eigenvector matrix. (For S to be invertible it is sufficient, but not necessary, that none of the eigenvalues be repeated.) The powers of A can be found by

$$A^t = S\Lambda^t S^{-1}$$

The total work vector U can therefore be expressed as

$$U = S\left(\sum_{t=0}^{T} \Lambda^t\right) S^{-1} u_0$$

If the magnitude of the maximum eigenvalue is less than one, then the design process will converge (i.e. as T increases to infinity the total work vector U remains bounded.) An eigenvalue greater than one corresponds to a design process where doing one unit of work at some task during an iteration stage will create more than one unit of work for itself at some future stage. Such a system is unstable and the vector U will not converge, instead growing without bound as T increases. (It is a sufficient, but not necessary, condition for stability that the entries in every row sum to less than one.)

A design process which does not converge would be one where there is no technically feasible solution to the given specifications, or one where the designers are not willing to compromise to find the technical solution. The remainder of the discussion is limited to problems where a technical solution exists and can be found in finite time.

The eigenvalues and eigenvectors of matrix A determine the rate and nature of the convergence of the design process. Much can be learned about what controls the iteration by looking at the eigenvalues and eigenvectors as opposed to looking at the sequence of remaining work vectors.[1]

Appendix B: Details on Delta Design Exercise

This appendix gives more detail about the design exercise. The exercise involves a group of four people working together on a design task. The object of the design is to construct a two dimensional structure made out of triangular red and blue elements (deltas).

Each of the four designers has one of the following roles: project manager, architect, structural engineer, and thermal engineer. The project manager is responsible for meeting cost targets, the architect is responsible for aesthetic considerations, the thermal engineer for temperature constraints, and the structural engineer for meeting the specified loads and moments.

The project manager is responsible for costs. Each element in the structure has a cost, and there is a cost for joining elements together. The cost functions are nonlinear.

The architect is responsible for aesthetic concerns. The goal of the structure is to produce a smooth exterior with a jagged interior (although these goals are not specified fully.) Also, it is desirable to use no more than 60% blue deltas.

The thermal engineer is responsible for thermal specifications. The temperature is a function of how many heat generating elements there are and how much radiating (exterior) surface exists. There are both local and overall constraints (maximum and minimum) on temperature.

The structural engineer is responsible for setting the points of attachment and checking the loads and moments. The point of attachment can support the

---

[1] The interpretation of the eigenvalues and eigenvectors for design problems is similar to the eigenstructure analysis used to examine the dynamic motion of a physical system. In dynamic system analysis, each eigenvalue corresponds to a rate of convergence of one of the modes of the system (a natural frequency.) The eigenvectors identify the mode shapes of natural motion, quantifying the participation of each of the state variables in each mode [Ogata 1967].

weight of than 20 deltas, although it is desirable to have a comfortable safety margin. Also, the structure must be capable of carrying internal moments, which occur due to cantilevering.

We have simplified the exercise slightly from its original version [Bucciarelli and Goldschmidt 1989]. We have removed the effects of variable gravity which are discussed, and we have simplified a few of the formulas. These changes have improved the clarity of the goals of the exercise to the participants, without making it a trivial design exercise. More details about the game (along with specific functional forms of the constraints can be found in [Gopal 1992].