Journal of Engineering Design

Publication details, including instructions for authors and subscription information:
http://www.tandfonline.com/loi/cjen20

Model-based Approaches to Managing Concurrent Engineering

Steven D. Eppinger [a]

[a] Massachusetts Institute of Technology, Alfred P. Sloan School of Management, 50 Memorial Drive, Cambridge, MA, 02139, USA

Available online: 24 Oct 2007

To cite this article: Steven D. Eppinger (1991): Model-based Approaches to Managing Concurrent Engineering , Journal of Engineering Design, 2:4, 283-290

To link to this article: http://dx.doi.org/10.1080/09544829108901686

Model-based Approaches to Managing Concurrent Engineering*

STEVEN D. EPPINGER

SUMMARY *As design managers begin to implement concurrent engineering in order to reduce development time, design procedures can become more complex, potentially taking even greater time to complete. This paper discusses the basis for such design task complexity and presents a method for representing these constraints within a design process model. These models are used to explore several approaches for design process management. It is shown that while the feedback characteristic of concurrent engineering is essential to enhance design quality, this feedback causes iteration which can use up valuable engineering time. For concurrent engineering to save time, we require a framework for evaluating which tasks are vital to begin early in the development cycle, and which tasks should be left for later.*

1. Introduction

Product development procedures have evolved from the traditional sequential scheme into concurrent engineering, where the manufacturing process is designed along with the product itself. As this transformation has occurred over the years, we find that the domain of product development has considerably broadened. As more life cycle concerns enter into the product development process, established design procedures become more complex and can even take longer to complete. This paper addresses the important challenge of performing product development projects more quickly in the face of many competing design concerns.

The approach taken in this research is to first understand the nature of design tasks as seen by engineers and their managers. To do this we create design process models using a suitable representation such as the design structure matrix, and we explore these models to understand the key challenges and tradeoffs in design management. Finally, we consider various approaches to managing such projects, and draw conclusions to formulate a design management strategy.

2. Design Management's Challenge

The development of a high technology product may involve hundreds of thousands of engineers making millions of decisions over the course of a few years. None of these

* An earlier version of this paper was published in *Proceedings of ICED '91, Zürich* (Zurich, Heurista).
Steven D. Eppinger, Massachusetts Institute of Technology, Alfred P. Sloan School of Management, 50 Memorial Drive, Cambridge, MA 02139, USA.

engineers performs in isolation [1]; rather, large and multidisciplinary projects involve a great deal of coordination among the engineering specialists [2, 3]. Indeed, this flurry of design activity cannot occur in a purely serial fashion, for it would take too long to execute the tasks one at a time. Just as we have learned with computing and with manufacturing, performing tasks in parallel speeds up the overall process, and this is the motivation for simultaneous or concurrent engineering [4]. However, in attempting to perform tasks in parallel, many engineers will find they are missing their requisite input information. If they were to wait until this information is available, they would be practicing sequential engineering. Instead, they must become involved in the creation of that information, and this interaction is the basis of design for manufacturing [5]. We find many manufacturing engineers today working closely with product designers. The problem becomes unmanageable when each engineer desires to work with all the others, since their responsibilities now overlap.

The design manager's challenge is to break down the very complex whole problem into many smaller subproblems that individuals or teams can tackle. If the problem is divided into subproblems that can be addressed entirely independently, then parallel (truly concurrent) engineering can occur. If, on the other hand, the subproblems are coupled, then how can engineers perform their work in parallel? Each engineer must wait for others to pass along information before his task can begin. In an information-transfer loop, someone must begin by guessing their inputs.

Dividing a large task into smaller problems is one of the most fundamental problem-solving paradigms; however, dividing the large problem effectively requires either a great deal of insight or an insightful model of the design process. In the next section, the design structure matrix representation and some associated analytical tools will be introduced for the purpose of modelling design procedures. In later sections, we will use design process models to help us discuss approaches to design management.

3. Models of the Design Process

To illustrate the need for design process modelling, we consider two design tasks, labelled A and B. Figure 1 shows directed graphs (digraphs) [6] of three possible ways in which the two activities can be related. If task B simply requires the output of task A (or vice versa), then the two tasks are *dependent* and would typically be done in series. On the other hand, the two would be entirely *independent* if tasks A and B could be performed simultaneously with no interaction between the designers. Finally, if task A needs information from task B, and also task B requires knowledge of task A's results, the the two tasks are *interdependent* or *coupled*.

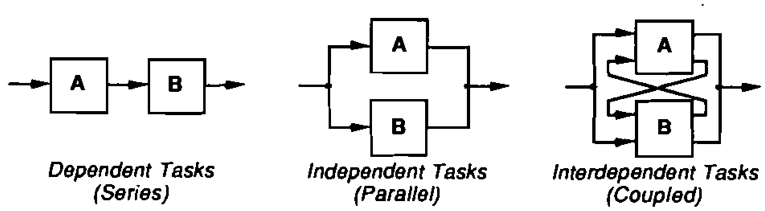

FIG. 1. Three possible sequences for two design tasks.

To coordinate either the dependent (series) tasks or the independent (parallel) tasks is quite straightforward. Certainly with no limitation in resources, the parallel tasks can be completed more quickly. The interdependent (coupled) tasks are much more challenging to organize, often requiring much more design time and many iterations of information transfer [7].

In the context of concurrent engineering, we can envision task A to represent a product design function, and task B to represent the associated manufacturing engineering function. Then our series model depicts the outdated 'throw the design over the wall' methodology. The parallel tasks model might then represent an idyllic view of simultaneous engineering, where both design and manufacturing functions are given the same challenge, and they magically develop product and process concurrently (without complex interactions). The coupled tasks model is a more realistic diagram of simultaneous engineering, where the information transfer is essential and iteration is typical.

Steward's design structure system [8–10] uses a compact matrix representation which allows the direct coupling of any task to another. Figure 2(a) shows the design structure matrix, where the design tasks to be performed are each represented by an identically labelled row and column of the matrix. The marked elements within each row identify which other tasks must contribute information for proper completion of the design. For example, the marks in row D are in columns E, F and L, indicating that completion of task D requires information to be transferred from tasks E, F and L. We would then desire these three tasks to be performed *before* task D. (The diagonal elements in the matrix are essentially meaningless at this point but are included to distinguish the diagonal and the upper and lower triangles of the matrix.)

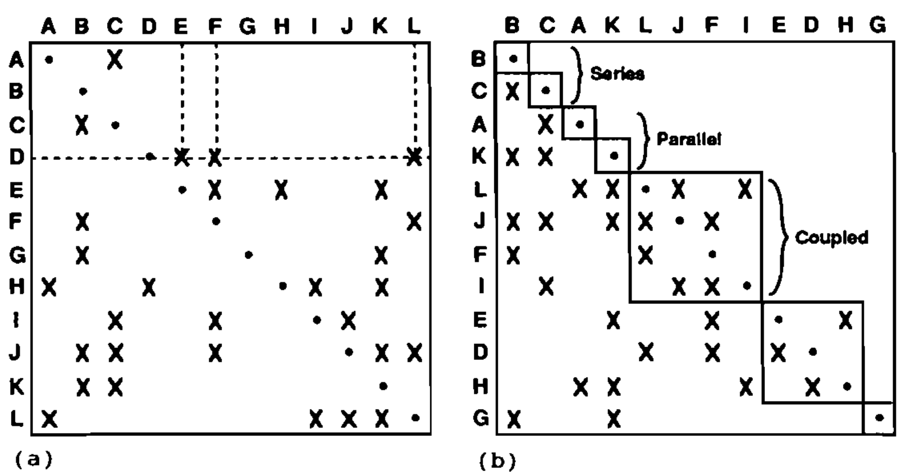

FIG. 2. The design structure matrix: (a) original matrix; (b) partitioned matrix.

Design structure analysis attempts to find a sequence of these design tasks which allows this matrix to become lower triangular. If the tasks can be sequenced so that each one begins only after it receives all the information it requires from its predecessors, then no coupling remains in the design problem. However, this rarely happens. Instead, analysis usually yields a matrix in block-angular form. Figure 2(b) shows the same matrix after the 12 tasks have been *partitioned* (rearranged) to achieve a more organized design sequence by interchanging rows and also swapping the corresponding columns.

The partitioning process [11, 12] has sequenced the tasks to be performed in the order: B-C-A-K-L-J-F-I-E-D-H-G. The matrix shows that task C is dependent upon task B, so they are partitioned in the sequence B-C. Tasks A and K can then be completed in parallel (since task K does not depend upon task A). The two 'blocks' encompassing the task sequences L-J-F-I and E-D-H identify two sets of coupled tasks, the most challenging aspects of this design problem. These tasks must be performed simultaneously, and the information transfer required may take the form of iteration and/or negotiation. We refer to the marks above the diagonal as *feedback*, since these represent later tasks providing input to the earlier tasks.

The partitioned matrix in Fig. 2(b) is not unique, but rather its form depends on the algorithm used to reorder the tasks. Several schemes for identifying the blocks are available, including techniques based upon binary matrix algebra [13], a rule-based (expert system) analysis [14–16], and Steward's loop tracing procedure [10, 17]. Still, we have developed improved partitioning algorithms which are discussed in another paper [11].

If the design structure matrix cannot be manipulated into lower triangular form, we then seek a form that minimizes the size and number of the remaining blocks on the diagonal. Collapsing these blocks into single tasks would certainly make the project *appear* to be simpler. In our example, we would combine tasks L, J, F and I into one task, and then collapse tasks E, D and H into another. We would be left with seven tasks in lower-triangular form instead of the 12 tasks as shown. However, this approach hides the real design problems and precludes any opportunity to further improve the design procedure by applying other techniques.

Since the coupled blocks in the design structure matrix represent design iteration, choosing the proper sequence to work through even these tasks is quite important. We believe that there is tremendous advantage in performing the initial 'guesswork' required to start the design iteration at a specific task, which may allow the design to converge quickly. This can reduce the time required by the iterative process by isolating uncertainty and increasing the confidence associated with the design decisions. Several algorithms also exist for sequencing within these blocks. Steward terms this procedure tearing, since guessing the unknown information corresponds to elements being torn from the matrix to get the iteration started. Effective tearing requires detailed knowledge of the problem domain so that the less important elements are torn to leave the essential ones below the diagonal. We are also developing improved tearing algorithms by modelling and analysing the design iteration in detail [18–20]. (Note that tearing does not actually alter the matrix by removing any of the marks; rather, these procedures simply find a suitable ordering for solution within a block.)

In our design management research, we have used the design structure matrix modelling approach several times. A more detailed description, extensions we have made to Steward's procedure, and data from several studies are to be found in our other publications [8, 21].

4. Comparing Design Structures

Using the design structure matrix as a modelling tool, a design team can consider various strategies for completing their product development task. In particular, they can assess how well the traditional sequential and the new parallel design approaches fit with the technical structure of their design tasks. Analysing the structure of a design procedure can identify many opportunities to improve the design process. As two examples of design improvement strategies, we present two conflicting approaches to consider: removing coupling versus adding coupling. These illustrate the important design-time/design-quality *trade-off* inherent in this decision.

4.1. Decoupling Tasks to Speed Design

This strategy attempts to accelerate the design process by removing some of the task coupling that is causing iteration. A coupled group of tasks can sometimes be split up into smaller groups by *artificial decoupling*, which involves actually removing one or more task dependences (one or more marks) from the matrix. This can be accomplished in several ways, including the creation of an additional task to be performed earlier or later in the design procedure. The definition of this new task would require the parties associated with the removed dependence to agree ahead of time on the relevant task interfaces. Another approach to this artificial decoupling strategy is illustrated by the following example which was obtained by comparing the design procedures in two firms developing competing products (an electromechanical instrument) [22].

Designers in one firm recognize three aspects of the product (the casing, wiring, and optics) to be so tightly coupled that they must be designed simultaneously, requiring lengthy negotiation (five to 10 design iterations, taking up to 6 months) before enough detail can be settled to build the first working prototype. The design structure matrix describing this procedure is shown in Fig. 3(a). The designers in the competing firm aim to deliver the first prototype much more quickly and believe that it is acceptable for the wiring inside such a prototype to be untidy. They have developed the design procedure illustrated by Fig. 3(b), where the wiring is absent from the design iteration loop. The design is completed more quickly (in two iterations, taking only a few weeks), and the prototype is built with crude wiring. The final wiring layout is completed for the second prototype. The wiring was artificially decoupled from the design in order to speed development.

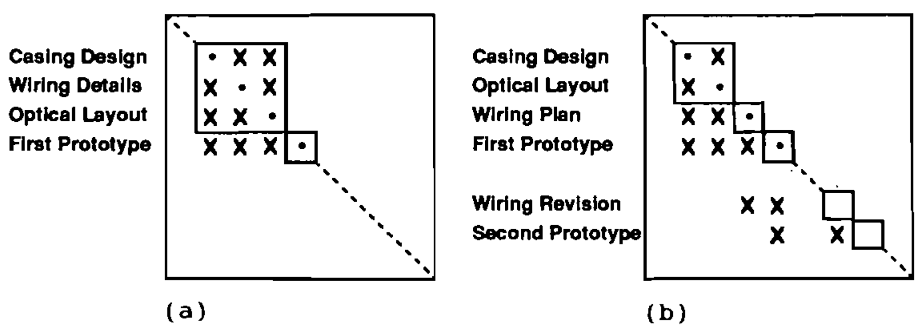

FIG. 3. Instrument design task matrices: (a) coupled; (b) decoupled.

4.2. Increasing Coupling to Improve Design Quality

The increased coupling strategy is the essential basis of concurrent engineering and design for manufacture (DFM). A portion of the traditional (sequential) process for the design of some engine components is depicted by the matrix in Fig. 4(a). The product designers perform their design tasks somewhat independently from the manufacturing engineers. In the modern (concurrent) design process, shown in Fig. 4(b), the practice of DFM mandates that these two activities be performed simultaneously. This is beneficial because the production expertise is brought into the early design stages (often causing much iteration), resulting in designs which are simpler to manufacture. However, the added coupling in the design process in fact slows product development considerably. Advocates of this philosophy would argue that overall design time can still be reduced because the need for later (more lengthy) iteration is therefore lessened. This is particularly true if the feedback from manufacturing engineering to design was indeed present in the original design procedure. This feedback is shown in Fig. 4(a) by the + signs which depict redesign activity addressing the production problems which inevitably arise.

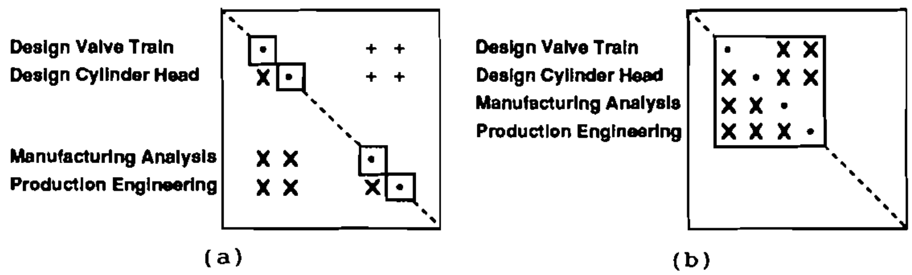

FIG. 4. Alternative design procedures: (a) sequential; (b) concurrent.

5. Conclusion

The discussion in this paper addresses the design manager's greatest challenge in the implementation of concurrent engineering: "how can we bring all of the important issues to the forefront of the design process without *slowing* the design procedure too much?" Placed in the perspective of the engineering team, "if we spend 80% of our engineering time in meetings, what will happen when concurrent engineering is implemented more broadly?" The answers to these questions are found by analysing the technical structure of the design domain.

The two scenarios presented in the previous section illustrate this fundamental trade-off facing engineering teams implementing concurrent engineering. One approach is simply to eliminate the manufacturing concerns from the early design stages of the project; however, since this scheme represents the traditional approach, we know that this sacrifices product quality. Other design management strategies involve either developing the manufacturing process in parallel with the design process or in a coupled iterative loop. The parallel approach ignores the coordination required, so quality also suffers in this scheme. The coupled design method improves quality but is often too slow. Obviously a hybrid strategy must be devised.

Some feedback of information is essential for quality or manufacturability of the product; other feedback is used to enhance the design only in the following revision or generation. Similarly, some of the feedbacks take a great deal of effort to facilitate, while others fit more naturally with the earlier design activities. Concurrent engineering succeeds in reducing overall design time *only when adding earlier iteration eliminates later iteration* which would have taken even longer. This is the basis on which the design-time/design-quality trade-off must be evaluated.

In summary, this paper presents a view of concurrent engineering that has not been adequately discussed in the literature. Concurrent engineering does not simplify the design process; rather it adds a tremendous amount of intertask coupling which makes the overall job considerably more difficult. Engineers and managers need a framework for modelling their design procedures and analysing any options that they find, such as the promotion of later tasks to the earlier stages of the design process.

5.1. Future Research

Most design research addresses trivial problems in great detail. Yet design managers are struggling with tremendously large and unstructured problems that defy rigorous analysis. The design struture matrix is a tool which allows groups to visualize the relationship among their various activities and reach consensus regarding which feedbacks are to be allowed. However, this tool does not actually show *how* to alter a design process, it merely provides a framework for analysing alternatives. The design team still must decide which feedbacks are worth including. Furthermore, the means of analysis of such alternatives is also unclear.

If several possible design procedures could be determined, then one should be chosen which minimizes design time and maximizes design quality. In this ongoing research, we are developing both models and analytical techniques which can assist in evaluating this trade-off. Our models are capable of predicting design time under a variety of iteration conditions, including probabilistic sequential iteration [18], deterministic work transfer [19], and purely sequential schemes [20]. Another approach we are taking is to explicitly consider the design process output quality in our models of design iteration [23]. This allows us to 'simulate' design procedures and to ask, as designers always do: "how can the process be improved?"

Acknowledgements

The author gratefully acknowledges intellectual contributions made by several collaborators in this research project: Daniel Whitney, Robert Smith, David Gebala, David Marshall and V. Krishnan. This research is funded jointly by several sources, including the National Science Foundation, the General Motors Corporation, and the MIT Leaders for Manufacturing Program, a partnership involving 11 major US manufacturing firms and MIT's engineering and management schools.

REFERENCES

[1] CLARK, K.B., CHEW, B. & FUJIMOTO, T. (1989) Product development in the World auto industry, *Brookings Papers on Economic Activity*, 3, pp. 729–771.

[2] LILEY, J.E.N. (1989) *The management of design, International Conference on Engineering Design, Harrogate, UK, August* (Zurich, Heurista), pp. 245–262.

[3] WOOD, K. (1989) *The organization and control of design for a large multidiscipline construction project, International Conference on Engineering Design, Harrogate, UK, August* (Zurich, Heurista), pp. 111–122.

[4] WHITNEY, D.E. (1988) Manufacturing by design, *Harvard Business Review*, July–August, pp. 83–91.

[5] DEAN, J.W. Jr. & SUSMAN, G.I. (1989) Organizing for manufacturable design, *Harvard Business Review*, January–February, pp. 28–36.

[6] WARFIELD, J.N. (1973) Binary matrices in system modelling, *IEEE Transactions on Systems, Man, and Cybernetics*, SMC-3 (5), pp. 441–449.

[7] SUH, N.P., BELL, A.C. & GOSSARD, D.C. (1978) On an axiomatic approach to manufacturing and manufacturing systems, *ASME Journal of Engineering for Industry*, 100, pp. 127–130.

[8] EPPINGER, S.D., WHITNEY, D.E., SMITH, R.P. & GEBALA, D.A. (1990) *Organizing the tasks in complex design projects, ASME Conference on Design Theory and Methodology, Chicago, IL, September* (New York, ASME), pp. 39–46.

[9] STEWARD, D.V. (1981) The design structure system: a method for managing the design of complex systems, *IEEE Transactions on Engineering Management*, EM-28 (3), pp. 71–74.

[10] STEWARD, D.V. (1981) *Systems Analysis and Management: Structure, Strategy, and Design* (New York, Petrocelli Books).

[11] GEBALA, D.A. & EPPINGER, S.D. (1991) *Methods for analyzing design procedures, ASME Conference on Design Theory and Methodology, Miami, FL, September* (New York, ASME).

[12] STEWARD, D.V. (1965) Partitioning and tearing systems of equations, *SIAM Journal of Numerical Analysis, Series B*, 2(2), pp. 345–365.

[13] LEDET, W.P. & HIMMELBLAU, D.M. (1970) Decomposition procedures for the solving of large scale systems, *Advances in Chemical Engineering*, 8, pp. 185–254.

[14] ROGERS, J.L. (1989) *DeMAID: a design manager's aide for intelligent decomposition user's guide, NASA Technical Memorandum 101575*, March.

[15] ROGERS, J.L. & PADULA, S.L. (1989) *An intelligent advisor for the design manager, NASA Technical Memorandum 101558*, February.

[16] SOBIESZCZANSKI-SOBIESKI, J. (1989) *Multidisciplinary optimization for engineering systems: achievements and potential, NASA Technical Memorandum 101566*, March.

[17] STEWARD, D.V. (1991) *Using information flow to manage the design of systems, Portland International Conference on Management of Engineering and Technology, Portland, OR, October* (New York, IEEE).

[18] SMITH, R.P., & EPPINGER, S.D. (1991) *A predictive model of sequential iteration in engineering design, MIT Sloan School of Management Working Paper no. 3160*, Revised August.

[19] SMITH, R.P. & EPPINGER, S.D. (1991) *Identifying controlling features of engineering design iteration, MIT Sloan School of Management Working Paper no. 3348*, October.

[20] KRISHNAN, V., EPPINGER, S.D. & WHITNEY, D.E. (1991) *Towards a cooperative design methodology: analysis of sequential decision strategies, ASME Conference on Design Theory and Methodology, Miami, FL, September* (New York, ASME).

[21] EPPINGER, S.D., WHITNEY, D.E. & GEBALA, D.A. (1992) *Organizing tasks in complex design projects: development of tools to represent design procedures, NSF Design and Manufacturing System Conference, Atlanta, GA, January* (Dearborn, MI, SME).

[22] MARSHALL, D.A. (1991) *Dynamic benchmarking: a comparative study of automotive suppliers*, Masters Thesis, MIT.

[23] GEBALA, D.A. & EPPINGER, S.D. (1991) *Modelling the impact of organizational structure on design lead time and product quality, MIT Sloan School of Management Working Paper no. 3301*, May.