INTERNATIONAL CONFERENCE ON ENGINEERING DESIGN
ICED 97 TAMPERE, AUGUST 19 - 21, 1997

REDUCING DESIGN CYCLE TIME AND COST THROUGH PROCESS RESEQUENCING

James L. Rogers

Keywords: design cycle, genetic algorithm, decomposition, concurrent engineering

1 Introduction

In today's competitive environment, companies are under enormous pressure to reduce the time and cost of their design cycle. One method for reducing both time and cost is to develop an understanding of the flow of the design processes and the effects of the iterative subcycles that are found in complex design projects. Once these aspects are understood, the design manager can make decisions that take advantage of decomposition, concurrent engineering, and parallel processing techniques to reduce the total time and the total cost of the design cycle. One software tool that can aid in this decision-making process is the Design Manager's Aid for Intelligent Decomposition (DeMAID).

The DeMAID software minimizes the feedback couplings that create iterative subcycles, groups processes into iterative subcycles, and decomposes the subcycles into a hierarchical structure. The real benefits of producing the best design in the least time and at a minimum cost are obtained from sequencing the processes in the subcycles.

2 Design structure matrix

Any nontrivial design project consists of numerous processes that are dependent on one another. This interdependency can become quite complicated. To reduce this complexity, numerous approaches, such as PERT, were developed to aid in understanding and managing these processes. Unfortunately, these types of tools are only applicable to sequential and parallel activities and cannot handle the iterative subcycles. A new tool was developed to manage and display the sequence of design processes in a complex design project and to handle the iterative subcycles that are commonly found in complex design projects. This tool, called the design structure matrix (DSM), was originally formulated by Steward [Steward 1981] and is the display format for DeMAID [Rogers 1996a]. A sample DSM is shown in figure 1.

	Process	Time	Cost
1	DYNMODL	30	30
2	STABDYN	40	20
3	STRMODL	10	50
4	STBQUAL	10	50
5	STRMODE	10	50
6	GEOMDEV	50	10
7	AEPROP	40	20
8	AEANALY	50	10
9	CTANALY	20	40
10	FLXAERO	20	40
11	INITDAT	40	20
12	RVSEDAT	30	30
13	MISPERF	30	30
14	VEHPERF	20	40
15	RGDAERO	30	30
16	AEROANL	20	40
17	PRESDEF	30	30
18	STRANAL	40	20
19	STRCTWT	50	10
20	WIANALY	40	20
21	AEROMDL	20	40
22	FINLDAT	20	40

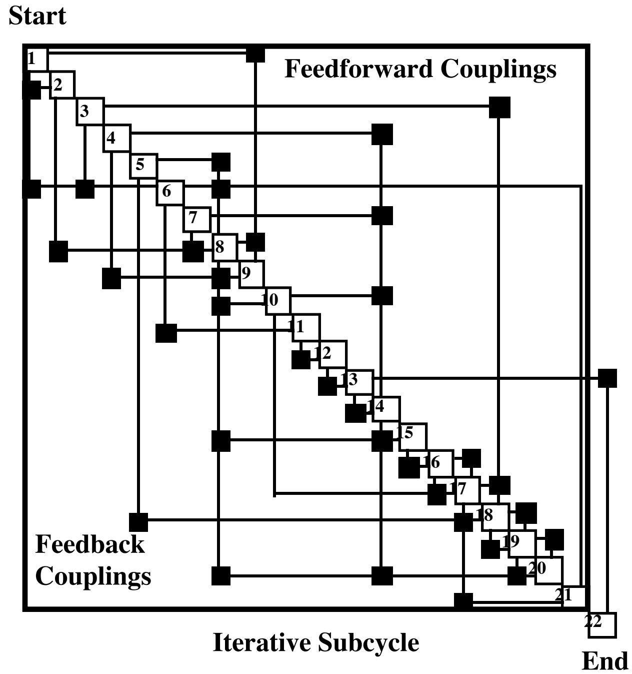

Figure 1. Design structure matrix

In the DSM in figure 1, the design processes for a representative conceptual design project are shown as numbered boxes on the diagonal. Execution begins at the top left box and proceeds to the bottom right. The process names and their associated time and cost are listed on the left. Output from a process is shown as a horizontal line that exits a numbered box, and input to a process is shown as a vertical line that enters a box. The off-diagonal squares that connect the horizontal and vertical lines represent couplings between processes. Squares in the upper triangle of the DSM represent feedforward couplings; squares in the lower triangle of the matrix represent feedback couplings. Feedback couplings indicate iterations in which initial data estimates must be made. Feedback couplings should be eliminated if possible; however, in many cases, not all of the feedback couplings can be eliminated. If certain feedback couplings cannot be eliminated, then DeMAID groups the processes into iterative subcycles. In figure 1, processes 1 - 21 are grouped into an iterative subcycle.

3 Hierarchical decomposition

After the iterative subcycles have been grouped, no feedback couplings remain outside the iterative subcycles [Rogers 1996b]. In this format, a problem, such as the generic design project shown in figure 2, can be decomposed into a hierarchy of iterative subcycles. In the decomposed hierarchy, iterative subcycles on the same level (e.g., blocks 3, 5, 6, and 9 in the figure) can be executed concurrently.

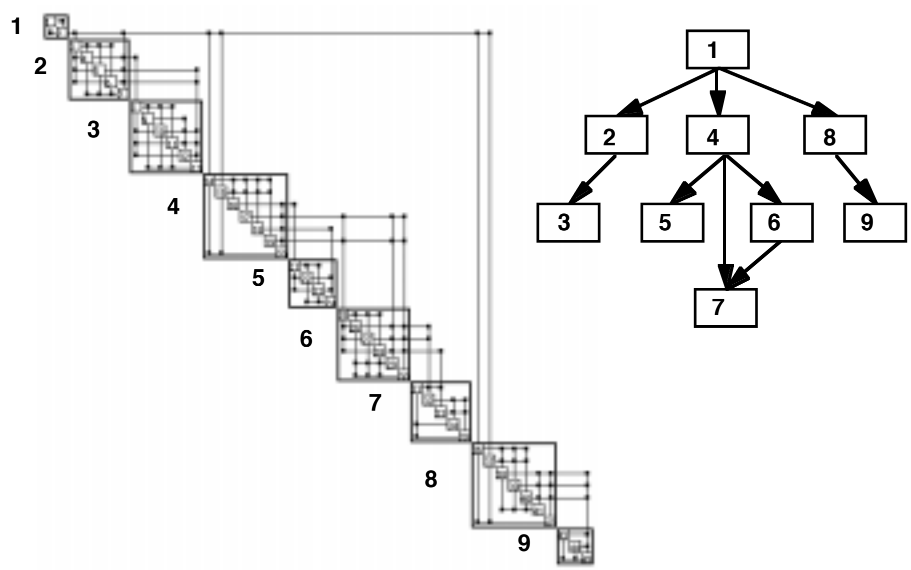

Figure 2. Hierarchical decomposition

4 Parallel processing within an iterative subcycle

Typically, a large design project will have several iterative subcycles as shown in figure 2. The DSM example in figure 1 is limited to a single iterative subcycle for illustrative purposes. It shows the process after the iterative subcycle has been determined but before the processes in the iterative subcycle have been ordered by DeMAID.

The hierarchical decomposition focuses on determining where concurrency exists based on iterative subcycles within the design cycle. Many iterative subcycles such as the one in figure 1 are large, and significant time savings can be achieved by executing some of the processes in parallel. To determine the parallel processing capabilities within an iterative subcycle, DeMAID assumes that all feedback coupling data are available as estimates [Rogers 1996b]. With this assumption, the processes in an iterative subcycle can be divided into a hierarchy in which any processes that fall on the same level of the hierarchy can be executed in parallel. Prior to reordering, the processes in the iterative subcycle for the DSM in figure 1 decompose into the following levels:

Level 1 - 1 2 3 4 5 6 7 10 11 12 13 15 16
Level 2 - 8 14 17 21
Level 3 - 9 18
Level 4 - 19
Level 5 - 20

5 Tracing a change in the design cycle

The DeMAID software has the capability to use the DSM to trace the effects of changes in the design cycle [Rogers and Barthelemy 1992]. A change made to one process, does not necessarily mean that all processes will be affected by the change. For example, refer to the DSM shown in figure 3. When a change is made to process 5, the design manager can see the affected processes (i.e., shaded processes must be reexecuted) with respect to process 29. The change to process 5 does not affect process 18; and the output from processes 23, 24, and 26 does not affect process 29. If any of these processes that do not require reexecution are costly, time consuming, or require a critical resource, substantial savings can be realized.

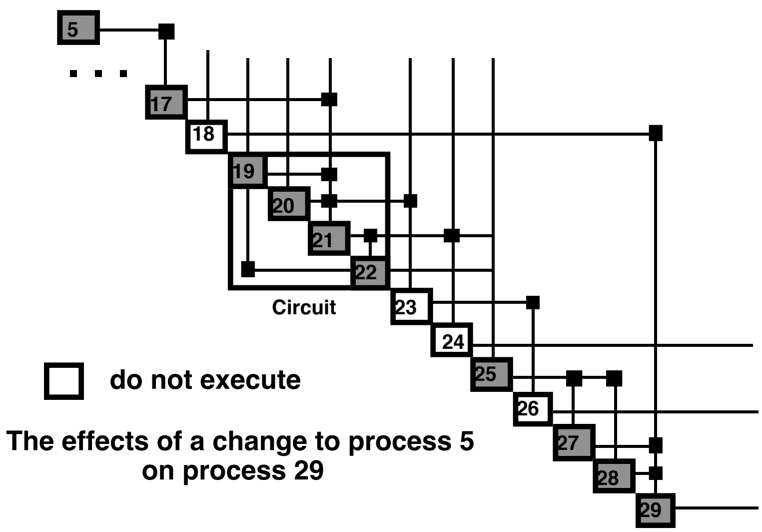

Figure 3. DSM for tracing the effects of a design change

6 Genetic algorithm for optimizing the design sequence

The primary advantage of the DSM format over earlier display tools is the capability to group and display the iterative subcycles that are commonly found in design projects. After the iterative subcycles have been determined, the processes should be sequenced so that the best design is produced in the least time at a minimum cost. The total cost and time to complete a design project are dependent on the sequencing of the processes in the iterative subcycles. A large, iterative subcycle, such as the one in figure 1, can be very expensive to converge because the iterations contained in the feedback loops are nested. For example, note in figure 1, there is a feedback loop that contains processes 8 - 15 indicated by the feedback coupling between process 15 and process 8. This loop is nested within another feedback loop that contains processes 5 - 18 indicated by the feedback coupling between process 18 and process 5. These nested feedback loops may require numerous executions of potentially expensive processes.

The DeMAID software contains a genetic algorithm (GA) that rapidly examines many different sequences and selects the optimum sequence of processes within each iterative subcycle [McCulley and Bloebaum 1994; Altus, Kroo, and Gage 1995; Rogers, McCulley, and Bloebaum 1996]. The fitness function of the GA minimizes the time and/or cost of each iterative subcycle. The estimated time and cost for each feedback loop are sums of the time and cost of the processes coupled by a particular feedback multiplied by an estimated iteration factor that is based on the strength of that feedback coupling [Bloebaum 1992; Rogers and Bloebaum 1994]. The estimates for each of the feedback loops are then summed to compute the total time and cost of the iterative subcycle.

The sequence of processes, selected for comparison purposes, in the DSM shown in figure 1 is not meaningful. This sequence has been optimized with the GA and is shown in figure 4. The original process numbers from figure 1 are displayed. With this reordering, the design-cycle time is reduced from 21,300 to 3,800 units; and the design-cycle cost is reduced from 19,640 to 3,220 units. However, with this ordering, much of the parallelism is lost (no more than two processes can be executed in parallel), thus the trade-off must be considered.

Process	Time	Cost
12 RVSEDAT	30	30
11 INITDAT	40	20
6 GEOMDEV	50	10
3 STRMODL	10	50
21 AEROMDL	20	40
16 AEROANL	20	40
17 PRESDEF	30	30
18 STRANAL	40	20
19 STRCTWT	50	10
20 WIANALY	40	20
5 STRMODE	10	50
15 RGDAERO	30	30
10 FLXAERO	20	40
8 AEANALY	50	10
2 STABDYN	40	20
1 DYNMODL	30	30
9 CTANALY	20	40
4 STBQUAL	10	50
7 AEPROP	40	20
14 VEHPERF	20	40
13 MISPERF	30	30
22 FINLDAT	20	40

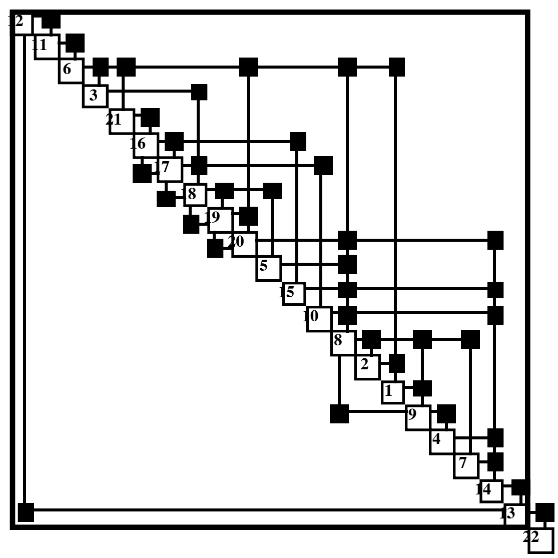

Figure 4. DSM with process sequence optimized by the genetic algorithm

7 Concluding remarks

The cost and time involved to complete a complex multidisciplinary design project can be substantially reduced by taking advantage of concurrent engineering, parallel processing, and decomposition techniques. To fully utilize these techniques and make the proper design decisions, the design manager must understand the flow of the processes within the iterative subcycles and optimize the flow of these processes. A tool such as the Design Manager's for Intelligent Decomposition (DeMAID) with both a genetic algorithm for optimizing the process flow and a design structure matrix format for displaying the iterative subcycles can aid in the understanding and thus in the decision-making.

References:

Altus, S. S., Kroo, I. M., and Gage, P. J. , "A Genetic Algorithm for Scheduling and Decomposition of Multidisciplinary Design Problems", ASME Paper 95-141, 1995.

Bloebaum, C. L., "An Intelligent Decomposition Approach for Coupled Engineering Systems", AIAA Paper No. 92-4821, 1992.

McCulley, C. M., and Bloebaum, C. L., "Optimal Sequencing for Complex Engineering Systems Using Genetic Algorithms", AIAA Paper No. 94-4327, 1994.

Rogers, J. L., and Barthelemy, J.-F. M., "Enhancements to the Design Manager's Aid for Intelligent Decomposition (DeMAID)", AIAA Paper No. 92-4809, 1992.

Rogers, J. L., and Bloebaum, C. L., "Ordering Design Tasks Based on Coupling Strengths", AIAA Paper No. 94-4326, 1994.

Rogers, J. L., "DeMAID/GA--An Enhanced Design Manager's Aid for Intelligent Decomposition", AIAA Paper No. 96-4157, 1996a.

Rogers, J. L., "DeMAID/GA User's Guide--Design Manager's Aid for Intelligent Decomposition with a Genetic Algorithm", NASA TM-110241, 1996b.

Rogers, J. L., McCulley, C. M., and Bloebaum, C. L., "Integrating a Genetic Algorithm into a Knowledge-Based System for Ordering Complex Design Processes", Proceedings of Fourth International Conference of Artificial Intelligence in Design, Stanford University, CA, 1996, pp. 119 - 133.

Steward, D. V., "Systems Analysis and Management; Structure, Strategy and Design", Petrocelli Books, Inc., New York, 1981.

James L. Rogers, Senior Computer Scientist
NASA Langley Research Center
Multidisciplinary Optimization Branch
M/S 159 Hampton, VA 23681-0001 USA
Phone: (757) 864-2810 Fax: (757) 864-9713
e-mail: j.l.rogers@larc.nasa.gov