JOURNAL OF AEROSPACE COMPUTING, INFORMATION, AND COMMUNICATION
Vol. 5, September 2008

Visual Dependency Structure Matrix for Multidisciplinary Design Optimization Tradeoff Studies

Kenneth W. English* and Christina L. Bloebaum†
University at Buffalo, Buffalo, NY 14260

DOI: 10.2514/1.32038

The design of complex products, such as automobiles and aircraft, necessitates modeling of the physics within numerous participating disciplines that are inherently linked through the transference of critical data. The resulting complexity makes it essentially impossible for designers or design teams, regardless of the number of participants or expertise of the participants, to intuitively understand the interdependencies of the disciplinary analyses and tasks. Recent methods and tools being developed in the field of multidisciplinary design optimization provide a means of representing these inherent couplings, as well as a way in which tradeoffs between accuracy and efficiency may be explored. It is with this in mind that the visual dependency structure matrix has been developed in this work. The visual dependency structure matrix is a web-based framework, coupled with underlying cost and error models, that enables designers to explore the possibility of eliminating or suspending couplings interactively, before or during a complex analysis run. The paper presents the visual dependency structure matrix framework and shows how the visual dependency structure matrix can be used as a mechanism for designers to cut computational costs during the design process without sacrificing the desired accuracy of the system level process.

I. Introduction

IN this age of exploding computational power, the design process is shifting almost entirely to the digital domain. Finite element analysis (FEA) and computational fluid dynamics (CFD) are just two examples of the wide variety of computer-based analyses that are currently used in the design process of large-scale complex products on a regular basis. Such complex products abound in many different industries, from a 777 jumbo jet to the microchips that drive the computers of the information age. The design of these complex products requires that the interactions of multiple analysis disciplines be appropriately represented to design a better product.

Researchers in the field of multidisciplinary design optimization (MDO) [1–5] are exploring innovative techniques for analysis and optimization that will result in more efficient, higher quality designs. MDO is a field developed specifically to address many of the computational and modeling issues pertaining to large-scale, complex system design including the inherent interactions in such systems. One major approach exploited in MDO is decomposing a large system into smaller subsystems, connected by information flows from the outputs of one subsystem to the inputs of another [1,6,7]. These information flows between subsystem analyses are termed couplings.

As an outgrowth of the field of MDO, computer codes have been increasingly connected together to allow for faster evaluation of possible designs. This connection of computer codes has provided an impressive decrease in

Received 8 May 2007; accepted for publication 20 March 2008.
* Deputy Director, New York State Center for Engineering Design and Industrial Innovation (NYSCEDII), kwe@buffalo.edu
† Professor, Department of Mechanical and Aerospace Engineering, MAE, clb@buffalo.edu

the time required to analyze a product, but also has an associated cost. Without the filter of a human in the loop, the role of experience and intuition has been greatly reduced in the process. The work presented within the current paper deals specifically with the issue of enabling designers to use their experience and intuition to best direct the computer-based analysis process in a large-scale design problem.

This work focuses on using the wealth of information available from the computer-based analyses to trade accuracy for efficiency in the optimal design process. Accuracy and efficiency are opposing drivers in developing design tools. Even with exponentially increasing computational resources, designers must decide whether to perform more detailed analysis of their designs, for instance, or to explore more design points. One example of this tradeoff is demonstrated in the aerospace industry, where the analysis process is so time-consuming that only one or two design concepts may be considered in the process of developing a new product.

With the increased computational and graphical resources available, visualization provides an increasingly attractive interface for designers as they develop new products. Through interaction with a virtual model, designers can return to the "sketch" phase, fully re-involving them in the design process, while including rigorous analysis in the background. This analysis must be able to keep up with the display of the virtual model, increasing the demand on efficient analysis. This demand for efficient analysis results in a need for designers to be able to trade accuracy and efficiency in the design process, as this tradeoff is deeply intertwined with the methods used to solve complex engineering problems. The concept of visual design steering was introduced as a way for designers to interactively steer the design process using visualization tools to increase efficiency [8–10].

The method presented here has been developed into the visual dependency structure matrix (VDSM), which enables designers to directly interact with their optimal design problem as the optimization progresses. This interaction takes place specifically through tradeoffs of accuracy for efficiency at the analysis stage. Using the VDSM, a designer can decide whether certain couplings are really critical, given that their representation might require a great deal of computational time, even though their impact on the system level accuracy is minimal. The approach includes the capability for a designer to visualize the semantics of a system in platform and geographically independent contexts.

II. Multidisciplinary Design Optimization

As the current engineering environment continues to evolve and available computing power increases, more details can be included in the analysis of each proposed design point in an optimization process. This more detailed analysis may now require multiple groups or computer codes to share information so as to evaluate the current optimal design point more accurately. The act of passing information couples two analyses together. At the same time, the demands for cost effectiveness and efficiency requirements increasingly restrict the optimal design process. Even with the potential for better designs resulting from the volume of available information, cycle times must be reduced. A group of approaches have been developed to address the solution of complex system design problems with these interrelated goals in mind. Within the field of MDO, methods have evolved to decompose these large, possibly unsolvable, problems down into smaller, more manageable subproblems. These subproblems are generally broken down by disciplinary area, and each can then be solved using different analysis tools and hardware [2,3,6].

The decomposition of these large system analyses results in a set of subsystems that are coupled together by the information passed between them [1,6,11–14]. One way to represent these coupled subsystems is shown in Fig. 1. Each of the 10 boxes along the diagonal denote participating coupled analyses (they could also represent tasks). Figure 1 shows that subsystem analysis 1 feeds data forward to subsystem analyses 4 and 6. Similarly, subsystem 3 feeds information forward to subsystem analysis 10. These couplings in the upper right hand side of the figure are all referred to as feedforwards, in this work. Feedforwards do not require any extra iteration, as all data are available when it is required for an analysis. However, subsystem analysis 1 requires data from subsystem analysis 9 before it can be implemented. This coupling, from 9 to 1, is termed a feedback, which represents data that feeds back upstream. This type of coupling, shown in the lower left part of the figure, represents iteration for the coupled analyses. These feedbacks, then, in turn, represent the drivers for computational expense during an optimal design process for a large-scale complex system.

As systems become larger, the number of subsystem analyses and time required to execute each can grow drastically. The interesting thing is that even though couplings may technically exist, they might not actually be "strong"

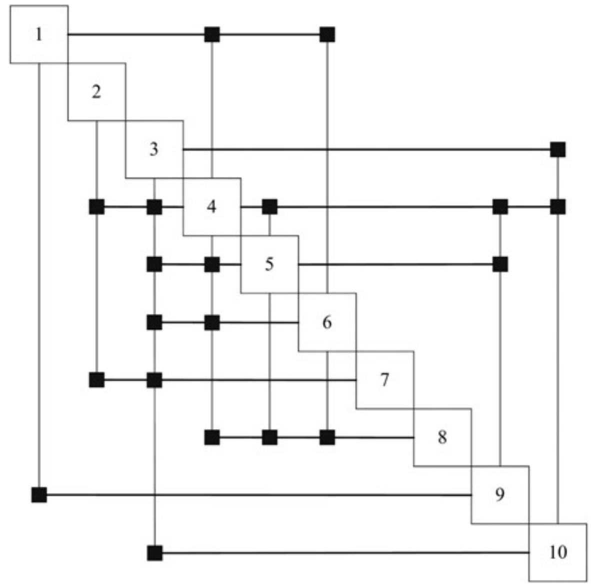


Fig. 1 DSM for coupled system.

enough to really impact the outcome of an optimization problem solution and, hence, final design. Consider an example system from aerospace design, with two subsystems interacting. The two subsystem analyses pertain to the aerodynamics of the wing shape and the wing structures groups. If the aerodynamics group changes the chord of the wing, this would have a considerable impact on the structures group. This is an example of a strong coupling. However, if the aerodynamics group changes the airfoil shape only slightly, that will most likely not change the internal wing structure. This is a weak coupling. The airfoil change may be an input to the structures group, but most likely would have a minimal effect on the end design. The passing of that information would most likely generate several design iterations, adding a great deal of unnecessary time and cost. However, if it were ignored, the result would have been approximately the same, with a greatly reduced cycle time and associated cost. This is the tradeoff that is addressed in this work, through the development of the VDSM.

In general, multidisciplinary engineering problems are most easily broken down into the disciplines that correspond to expertise in specific areas [15–18]. It has now become accepted that the traditional over-the-wall approach to design of large-scale coupled systems, in which an order was imposed for the disciplinary designs, with one completing and then passing their design to the next discipline in line, resulted in suboptimal designs that were order dependent (i.e. the designs drastically changed depending on the order of the implementation). This hierarchical design process has no way to leverage the interdisciplinary interactions, but rather tries to overcome them through brute force.

Sobieski presented an approach in the late 1980s termed the global sensitivity equation (GSE) method to find total derivative information that captures the impacts of coupling [12]. These total system level derivatives can be used in a number of ways to construct a system level optimization problem, using the derivatives to create approximations of the desired outputs [11,14]. This approach takes advantage of the benefits resulting from sequential linear optimization without losing a significant degree of the subsystem interaction information. While the sequence of subsystem analyses is still important, the true impact is on computational cost and not on final solution efficacy. If a sequence of subsystems has no feedbacks, then that sequence only needs to be executed once for the output variables to be converged. In cases with feedbacks, iteration is required. The more feedbacks a system has results in more iteration and increased computational cost.

A. Coupling Strength Analysis and System Reduction

To perform a tradeoff of accuracy and efficiency in the optimization of a complex system, some basis for comparing options must exist. As discussed in the previous section, the design and analysis of a monolithic complex system can be decomposed into a set of interdependent subsystem analysis problems with an overall system level optimization problem. This represents one approach for performing a system optimization and is termed the multidisciplinary feasible (MDF) strategy [19–21]. The set of analyses are linked by couplings that transfer data from one subsystem to the next. To study the impact of these couplings further, some method of quantifying their strengths must be identified. Two principal methods have been developed: local sensitivity-based and total derivative-based [22,23]. These methods differ in the way in which the strength of a coupling is measured. Local sensitivity methods focus on the subsystem to subsystem interactions only, while total derivative-based methods focus on the impact couplings have on the overall optimization problem (that is, objective function and constraints).

Owing to the high levels of inter-connectivity encountered in complex systems, a method of evaluating coupling strengths is important to finding ways of reducing computational costs in the analysis and, subsequently, optimization process. Computational savings can be achieved through elimination of couplings in two ways. First, savings is realized because the number of subsystems that need to be analyzed to converge the subsystem outputs at the beginning of the optimization process may be reduced. Second, if couplings are suspended or eliminated from the system, then the computationally expensive sensitivity information relating to those couplings does not have to be computed.

In system reduction, designers use coupling strength information to identify and then temporarily remove weak couplings between subsystems [22]. After solving the GSEs, the normalized local sensitivity information is used as a measure of a coupling's strengths. Using heuristics, the designer can identify couplings that have little to no impact on the system level optimization problem. These couplings are considered weak, and can then be chosen for elimination or temporary suspension from the problem.

Bloebaum and Rogers incorporated the local coupling strength determination in a methodology to determine the impact a local sensitivity has on a constraint or objective function [22,24]. The problem with this approach is that it is possible for a coupling to have a small local sensitivity, but have a significant impact on the end result of the optimization problem.

The total derivative-based coupling sensitivity analysis approach was developed to address the needs of coupling strength analysis in a complex system optimization context [23,25]. Miller developed a method that provides information on the error introduced into the objective function and constraints when a certain coupling is suspended for a single cycle [25]. In this method, each coupling is considered for suspension in every cycle, including both the error introduced and the potential savings by suspending that coupling. Using this approach, every coupling is free to be suspended or active every cycle, resulting in a binary problem of choosing which couplings to suspend for each MDO cycle. The computational cost of solving this problem must be limited to prevent it from exceeding the resulting savings. The total derivative model was then extended to account for the propagation of error through multiple cycles of suspension [23]. Multiple-cycle coupling suspension requires a more complex sensitivity model, because when a coupling is suspended the sensitivity information that is used to generate the error introduced is not available for the new design point. These sensitivity-based coupling strength metrics are particularly important, as both are represented in the VDSM developed in the present paper.

B. Coupling Suspension Selection Problem

The local and total derivative based coupling strength methods provide an estimate of the introduced error resulting from suspension of couplings. This estimate of the propagated error allows the designer more flexibility in choosing to reactivate or leave couplings suspended. An optimization subproblem can then be created using this information, for the purpose of identifying the best set of couplings to suspend, without exceeding error limits imposed by the designer.

As previously stated, suspending a coupling results in a reduction in computational time for both the sensitivity analysis calculations and system convergence stage. The computational savings pertaining to the calculation of finite differences can be calculated relatively easily, but the system convergence savings is more difficult to formalize. Since the convergence time increases as the number of feedbacks in the problem increases, a designer may wish to target feedbacks for suspension. This approach can be seen in Fig. 2, where the basic MDF approach is being implemented. After initializing design variables, the coupled system analysis is performed, followed by the sensitivity analysis (using the finite difference method or the GSE method, typically), followed by an approximation based optimization of the system level objective function and constraints. Once this cycle has been implemented at least once, a coupling

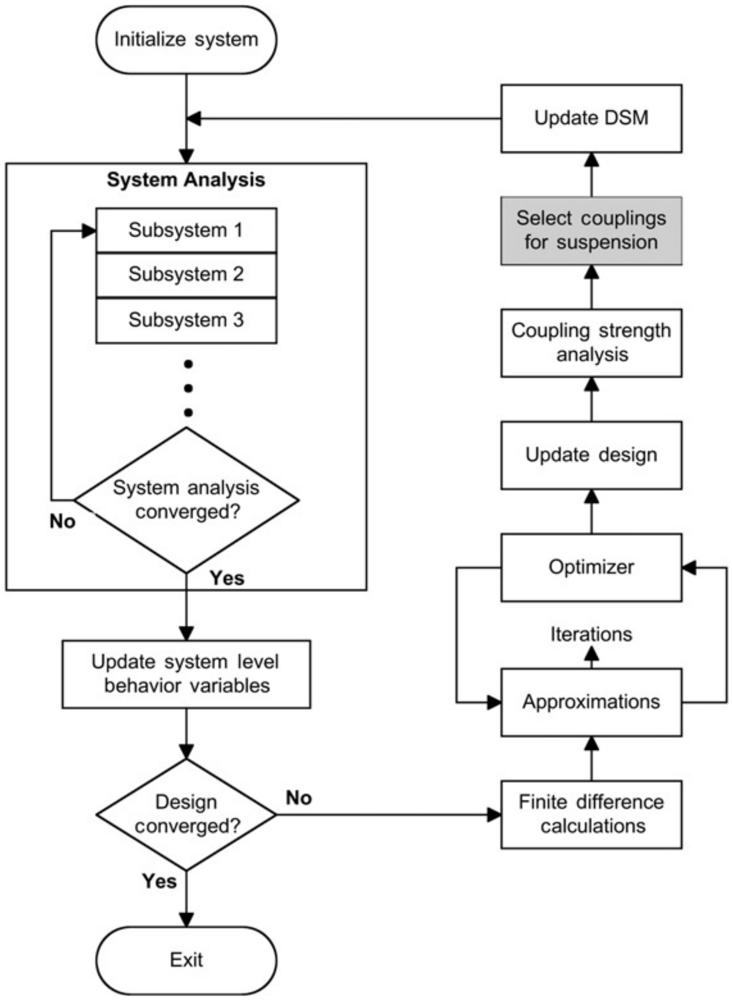


Fig. 2 MDF approach with coupling suspension.

strength analysis can be performed, along with a determination as to whether any couplings should be suspended before the next cycle.

In English et al. [23], an optimization problem is developed that maximizes a savings function subject to constraints ensuring that errors in the system level objective function and constraints (due to suspension of a coupling at the analysis stage) do not exceed prescribed allowable errors. The optimization problem in Eq. (1) then results.

$$
\begin{aligned}
\text{Maximize} \quad & \sum_{i=1}^{I}\sum_{j=1}^{J}\left(\text{Savings}_{A_{ij}} \cdot \delta_{A_{ij}}\right) \\
\text{Subject to} \quad & \sum_{i=1}^{I}\sum_{j=1}^{J}\left(F_{\text{error}_{A_{ij}}} \cdot \delta_{A_{ij}}\right) \leqslant F_{\text{error allowable}} \\
& \sum_{i=1}^{I}\sum_{j=1}^{J}\left(G_{k_{\text{error}_{A_{ij}}}} \cdot \delta_{A_{ij}}\right) \leqslant G_{k\,\text{error allowable}} \\
& \qquad\qquad\qquad k = 1, K
\end{aligned}
\tag{1}
$$

In this optimization subproblem, the Savings_{ij} term is the savings owing to suspending coupling_{ij}. The terms $F_{\text{error}ij}$ and $G_{\text{kerror}ij,}$ are the errors introduced into the objective function and constraints by suspending coupling_{ij}. The δ_{ij} term is a binary operator that can either have the value of 0 if the coupling is active, or 1 if the coupling is inactive. Here, $F_{\text{allowable error}}$ and $G_{\text{kallowable error}}$ are the designer specified error limits. The goal of this embedded optimization problem is to find the set of couplings that will yield the greatest time (or cost) savings, without exceeding the allowable error limits on the analysis. This embedded optimization problem must be solved during every MDO cycle to find the best set of couplings to suspend for the following cycle.

There are two disadvantages to implementing this coupling suspension problem. The first is one of computational impact, given that the introduction of this optimization subproblem adds computational time to every cycle. The second, and perhaps most important, is that the coupling suspension problem (CSP) is automated, thereby not enabling the designer to use knowledge, experience, or intuition to make decisions about what couplings are critical or not. The VDSM has been developed so as to allow the designer to be an active participant in these decisions. To best understand the development of the VDSM, it is important to have some background on existing approaches in the field of design process management. This background is provided in the next section.

III. Design Process Management

Design process management pertains to the interaction of a designer or manager with the progress of the solution of a design problem. The roots of this process lie in graph theory [26]. The expansion of graph theory to project management resulted in the development of tools such as the PERT chart [27], which provides important information for tracking the progress of well-defined design and production tasks. An important limitation to note for PERT charts is that iteration cannot be incorporated into the chart. A variation on PERT, the general evaluation review technique (GERT), can handle cycles effectively for simple networks [28]. Another primary technique for conveying design process connectivity is the process flow chart. This chart is the familiar directed graph, with labels instead of numbers on each of the nodes. As process flow charts become more complex, however, it becomes more difficult to understand the sequence of tasks to be executed

All of these methods are easily scalable and stable, allowing their use on very large design projects. These methods also apply very well to the traditional over-the-wall sequential design approach. As each group completes a task, progress is marked off, and the next group is given approval to proceed. As products have become more complex and analysis more detailed, however, these approaches have encountered some difficulties in representing the complexity of the system in an easily accessible manner.

A. Dependency Structure Matrix (DSM)

The design structure matrix (DSM), also referred to as the dependency structure matrix, was developed by Steward as an alternative to the process flow chart [29]. A sample DSM is demonstrated in Fig. 1. The modules or subsystems are represented as numbered boxes along the diagonal of the DSM. Recall that the output from modules is represented by horizontal arcs, while vertical arcs represent the input to the modules. The connectivity between the output of one module to the input of another is represented by a black square symbol. This symbol represents the coupling between the modules. Here, couplings in the upper diagonal of the DSM represent feedforward couplings, implying sequential execution. Couplings in the lower diagonal of the DSM represent feedback couplings, implying iteration. It should be noted that some researchers use a reverse representation for feedbacks and feedforwards. Iteration results from the need to estimate input from other modules before it is available. This cyclic behavior is clear when using the DSM. Additionally, the DSM may be used to group iterative subcycles that may appear in a design process (termed circuits), making the DSM scalable to larger design processes.

One of the major strengths of the DSM is that it lends itself easily to the rearrangement of the participating subsystems to reduce iteration or overall system execution time and cost. Successful explorations of module sequencing and execution have been made, resulting in both theoretical sequencing advancements and formal design management tools. Research has demonstrated that sequencing the modules in a DSM based on time and cost considerations, as well as the strength of the couplings between modules [24,30–33], can substantially impact the time and cost of the process. Eppinger et al. have also advanced the theory of the representation of the links between modules, using a purely heuristic-based approach [34]. There has been much research in the use of the DSM for various design management functions over the past several years by researchers in many fields, and particularly in the aerospace field, including Rogers [35,36], Krishnan et al. [37–39], Pimmler and Eppinger [40], Kusiak and Wang [41–43], Kusiak et al. [44–49], Guo et al. [50], McCulley and Bloebaum [31,32], Eppinger et al. [34,51–53], Gebala and Eppinger [54], Cho and Eppinger [55], Braha [56], Browning et al. [56–60], and Sosa [61]. These developments have resulted in multiple design management tools based on the DSM, the most significant of which are discussed in the next section.

B. DSM-Based Design Management Tools

A comprehensive overview of the DSM design management tools is reviewed in Smith and Eppinger's 1997 review of iteration control mechanisms and Rogers' 1999 reviews of decomposition tools and techniques [62,63]. DSM-based tools can be used by two differing audiences. The first is that of design managers interested in the frequency and content of information exchange, as well as the resulting iterative effort. This use of the DSM can use "soft" information, including probabilities of exchanging data, and the percent of the original effort that is required if iteration results. Engineering uses of the DSM result in tools more focused on the transfer and representation of the numeric values associated with the module and coupling semantics. In the engineering use, especially when the DSM represents a number of linked analysis codes, each module often requires full execution of its analysis, even if the change in the input to that module is small. This characteristic drives the importance of representing the semantics of the couplings between modules.

The problem solving matrix (PSM), developed by Steward, drives the organization of teams based on the problem structure, instead of the institutional structure of the designing entity [64]. The PSM technique of process management relies on two decision-making stages for the design process topology. The first phase, partitioning, reduces the matrix into a set of iterative subcycles. The second phase then removes couplings and, subsequently, tests to see if the iteration still exists. Coupling strengths are typically anecdotal, but may also use numerical data. The designer then selects which couplings to permanently remove from the system through "tearing". The act of tearing relies on the design manager's judgment and experience, as these links are permanently removed from the analysis process. Other tools have been developed that allow designers to use matrix-based tools to gain insight into the effort required to converge the analysis phase of the design process, using the expense of each analysis module, and some representation of additional work required upon iteration.

The work transformation matrix (WTF) is the result of significant research at the Sloan School of Management at MIT [52,62,65]. The WTF is used to predict convergence characteristics of a certain system topology [65]. The WTF contains a matrix of data of two differing types: along the diagonal of the matrix are the costs or time required to evaluate a module while the off diagonal elements represent the percentage of the initial effort that is required if that coupling has information flow through it. Through analysis of the off diagonal matrix of the WTF, researchers have developed techniques to identify the nature and rate of convergence for a given DSM.

A widely used DSM-based tool is the Design Manager's Aide for Intelligent Decomposition (DeMAID) [35], developed at NASA-Langley Research Center. DeMAID is a computer tool that allows a designer to easily implement partitioning and sequencing strategies on a system topology. The system can be initially partitioned in a manner similar to Steward's. Iterative cycles are identified by enclosing squares, called circuits [30]. Extensive research has been performed to integrate intelligent sequencing of modules into DeMAID [30]. Upon the completion of the tearing operation, an optimal process sequence is found using a robust optimization tool [31,32]. This tool can optimize the system topology for optimal solution time, cost, or a combination of both, using an incorporated genetic algorithm [63,66].

Coupling strengths in DeMAID's DSM are color coded on the basis of local sensitivity, and are broken into seven levels of coupling strengths [24]. These strength levels can be determined statistically or through user input. DeMAID also allows for a tracing the impact(s) of a design change. DeMAID can display what modules are impacted if a design variable or module output changes [66]. Finally, DeMAID can identify potential opportunities for the parallel execution of modules.

Other researchers have explored alternative uses of the DSM. Kusiak and Wang relate design variables and processes with a DSM-like incidence matrix [41–43]. The incidence matrix can be used to improve the decomposition of a multiple subsystem design process. Michelana and Papalambros use the functional dependence table (FDT) to represent the connectivity between variables and the objective function and constraints [67]. Heuristics are then used to decompose the design problem into multiple subproblems. Liu and Brown [68] and Kroos et al. [69] offer other decomposition techniques, using variants of the DSM. A commercial DSM tool, Planweaver, was released in 2001, and provides a commercial implementation of two-dimensional DSM approaches primarily for architectural and construction applications [70,71]. It collects the dependency information and structures the data in a form to aid design and construction teams in assessing risk to their projects. Other DSM planning tools, including DeMAID [35,66], PSM32 [64], Multiplan [72], Acclaro Designer [73], and Lattix LDM [74] incorporate some enhancements to the basic digraph representation of the DSM, using either a binary matrix filled with zeros and ones or a two-dimensional block diagram representation of a DSM. However, in order for designers of complex products to consider as much information as possible in their decisions, it is critical that more cues be included to condense information into a useful format for designers.

To summarize, then, a wide range of design management tools exist to help manage complex system topologies. While most are rooted in graph theory, each has its own unique appearance and purpose. A number of different measures exist for designers to use in considering the importance of couplings, and their impact on the iteration and convergence characteristics of a design process. The VDSM uses the DSM concept as a basis, but includes representation of sensitivities, cost for implementation of subsystem analyses, time for subsystem analyses, couplings, and a metric pertaining to impact of coupling suspension on error of desired attributes at the system level.

IV. Visual Dependency Structure Matrix (VDSM)

Previous work in managing complex analyses has primarily focused on managing the solution of analysis or optimization problems. Some visual tools allow managers to more efficiently execute the optimization and analysis processes under their control. Tools developed for this purpose focus on proper sequencing of the analysis modules or on reducing the quantity of data passed between modules. This decrease in data passing, also termed system reduction, must take into account many factors to trade-off analysis accuracy with computational efficiency. Previous methods allowed designers to alter the topology of their analysis process using only a reduced set of the available information. This paper presents an approach that provides designers an intuitive interface to better understand the topology of the analysis phase of a formal optimization process. This approach synthesizes information, including analysis module times and costs, as well as local and global coupling strengths, into a visual format. The interface is implemented on a sample system, demonstrating the use of visual cues to allow designers to interact with their problem. The introduced approach is platform and geographic independent, and highly extensible.

This work focuses on a single phase of the overall MDO process, and uses the initial implementation of the coupling suspension problem as its starting point, allowing a designer to use intuition and experience to account for the factors included in the coupling suspension problem (recall Fig. 2 and Eq. (1)). By developing a technique that allows a designer to interactively and intuitively select couplings for suspension, designer intuition and experience can be reincorporated into the design management process for large-scale complex design problems. The coupling selection takes place in the highlighted section of Fig. 2. To interact with the topology of the analysis phase of the optimization process, the designer must consider all of the factors mentioned previously (for example, time, cost, error impact, and so on). Most critically, the representation of these factors must be meaningful, so the designer can easily make choices on how to sequence and remove couplings. The problem is that some of these metrics result in extremely large amounts of data, even for relatively small problems. This fact is illustrated for a simple five subsystem problem, as shown in Fig. 3.

The problem structure, including the sequencing and subsystem couplings, is quite evident from this figure. However, when using a DSM representation such as this, it must be augmented by additional data. The data in Table 1 must be used for the designer to form an idea of the time and cost of each analysis module. Additionally, if the designer wishes to rank the couplings for consideration of removal based on their impact on other subsystem behavior variables, Table 2 must be used. If the designer wishes to also include the impact of suspending couplings on the optimization problem in the system reduction process, even more information is required, as in Table 3.

When selecting couplings for suspension, the designer must balance the benefits resulting from suspending a coupling and the resulting decrease in solution quality. The benefits of suspending a coupling can be estimated using the DSM and Table 1. The impact of suspending a coupling on solution quality requires more consideration. Designers must account for the impact suspending a coupling will have on both the analysis and the optimization

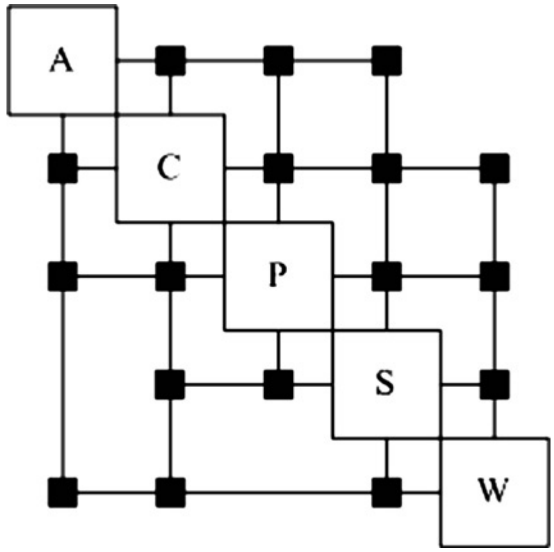


Fig. 3 DSM for five coupled subsystems.

Table 1 Subsystem time and cost

Module	Time	Cost
A	250	75
C	300	50
P	400	150
S	150	250
W	200	100

Table 2 Local coupling sensitivities

From module	To A	To C	To P	To S	To W
A	—	VS	W	VS	—
C	W	—	VW	S	N
P	N	VW	—	N	N
S	—	N	VW	—	VS
W	S	W	-	N	—

Coupling key: VW: very weak;W:weak; N: nominal; S: strong; VS:very strong

problem. The impact on the analysis can be estimated using Table 2, which will not overwhelm a designer for small problems, but can be cumbersome to manage. One can easily imagine, however, how difficult it would be for a designer to make sense of such tables for a 20 or 30 module system, let alone for a 100 module system.

Table 3 shows the impact that suspending a coupling will have on the system level optimization problem. To trade off accuracy in the optimization problem with computational efficiency, designers must understand the error that suspending a coupling will introduce into both the objective function and constraints. One must recall that the introduced error is used in the coupling selection problem that is posed in Eq. (1). This combinatorial optimization subproblem must be solved by the designer to find the best set of couplings to suspend. Taking all of these tables collectively in considering the coupling selection problem, it is clear that this approach, using hard data together with the DSM, is cumbersome for a designer to use, even for a small five subsystem problem.

Table 3 Coupling impact on system level measures

Coupling (from–to)	F	G1	G2
A–C	L	VL	VL
A–P	S	VS	VS
A–S	N	S	N
C–A	VS	VS	VS
C–P	VS	S	VS
C–S	VL	VL	VL
C–W	VS	VS	VS
P–A	S	L	N
P–C	N	N	VL
P–S	S	S	S
P–W	L	L	VL
S–C	VS	VS	VS
S–P	S	N	N
S–W	N	VL	L
W–S	VS	VS	VS
W–C	S	N	N
W–A	N	VL	L

Error key VS: very small; S: small; N: nominal; L: large; VL:very large

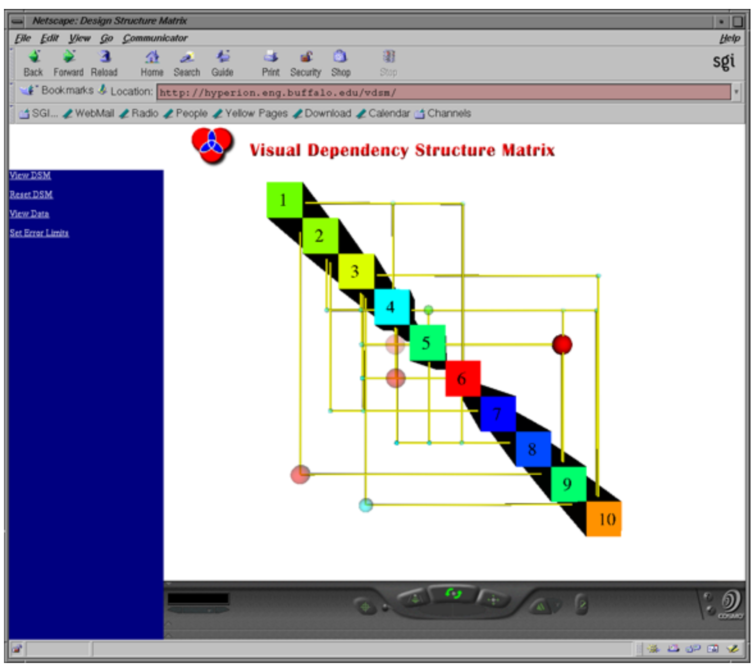


Fig. 4 VDSM for 10 coupled subsystem problem.

The VDSM developed in this work is shown in Fig. 4. The goal in developing the VDSM is to represent the tremendous amounts of data resulting from a large-scale complex design problem, in an easy to understand and intuitive fashion, using color, depth, and transparency, so as greatly to assist the designer in the management of the analysis phase and the solution of the coupling selection problem.

The needs developed in the previous section have been addressed through the development of a web-based tool to manage the solution of complex system optimization problems. This tool uses visualization to incorporate all of the various data that describe the structure and sensitivity characteristics of the analysis phase of the optimization problem. The goal of this development is to allow designers to view and interact with the model of the system topology, so as to allow a designer to consider much more complex systems, intuitively and easily.

The methodology developed relies on a platform and geographic independent environment, developed with the internet in mind, specifically the worldwide web (WWW). Many platform and geographically independent tools have been developed using the WWW as an enabling technology [8–10]. Development of an application for deployment on the web implies an adherence to platform independence. Web browsers are used here as the client application on a user's computer to visualize the system topology and data in the VDSM.

In the VDSM, the data and information are conveyed in a way that enables designers to include the third dimension of depth into the display, so that more cues are available for designers, allowing them to intuitively understand more information about their design process. This three-dimensional representation is non-immersive and includes cues other than depth. Stress analysis tools such as ANSYS, NASTRAN, and ABAQUS use a color variation scheme to convey differing levels of stresses in a component. With extremes of blue and red, and a variation between these extremes, designers can quickly and intuitively view a three-dimensional stress analysis result and inspect it for areas of high concentration. The VDSM incorporates a color coding scheme such as this for intuitive representation of variations in the data.

In the VDSM, the connectivity between modules can easily be displayed using a familiar DSM-like interface. Modules have two primary metrics: how much the module costs to execute, and how long it takes to execute. These characteristics can easily be accounted for in some combination of size and color. Here, time is represented by color variation associated with each module, while cost is represented by the depth of each module. The nature of the couplings between modules presents a more difficult set of obstacles to navigate. Considering a sample complex system with p behavior variables and m constraints, it is clear that for each coupling associated with the p variables there are $m + 2$ unique characteristics. These characteristics are:

1) 1 – local coupling sensitivity
2) 1 – % error introduced into the objective function if the coupling is suspended
3) m – % error introduced into each constraint if suspended

The local coupling sensitivity indicates to a designer how much an impact one subsystem's output will have on another subsystem's output. In simpler terms, it gives an indication of how much of a coupling exists directly between the two modules. We have chosen to intuitively represent this "existence" measure by using a visual cue of transparency. Hence, if a coupling strongly connects two modules, it would be completely opaque. If the coupling only very loosely couples the systems, the coupling could be represented as very transparent.

The next type of data to be addressed pertains to percent error associated with a coupling's existence or lack thereof. There are two obvious sets of data associated with percent error. One is the percent error introduced into the objective function. In the VDSM, this is easily be represented by a single visual cue: size of the coupling. The second set of data is the percent error introduced into the constraints when a coupling is suspended. There can be anywhere from zero to thousands of constraints in a design problem, so some flexible strategy must be developed to convey this information. This reduces to a simple decision of whether to visualize the average error associated with all these constraints or the maximum error. Considering that a designer is concerned with the impact that suspending a coupling would have on the overall optimization problem, the use of a color variation cue to display the relative size of the error introduced into the problem appears logical, especially as the coupling strength analysis assumes normalized constraints. An alternative format would be to consider the percent error the coupling introduces into each constraint and to then select the maximum value. While this would be excellent for highlighting points of concern, the broader issue of impact on the overall design problem would be lost, as no indication of the impact the coupling would have on other constraints would be conveyed. Hence, in the VDSM, the choice is made to reflect the average percent error in the visual cue, through the use of color of the coupling. As an overview, therefore, the VDSM uses the following visual cues for the unique characteristics below

1) local coupling sensitivity—coupling transparency
2) % error introduced into the objective function if the coupling is suspended—coupling size
3) average % error introduced into constraints if suspended—coupling color

Now that the visual cues used in the VDSM have been discussed, other aspects of representation must be considered. Any interface developed must allow designers to view the underlying data used to develop the display. This is not only important for troubleshooting questions about a model's accuracy, but also may be used by designers when impacts of specific couplings on specific aspects of the overall process are required. Another goal of the interface is interactively to allow designers to investigate the impact that design process management decisions will have on the overall design. As such, some means of interaction with the model must exist. Designers require the capability to select couplings for removal or reactivation, while receiving feedback on the impact this has on the design process. This type of desired interactivity drives the choices for the languages and technologies to be used in the development of the VDSM.

The desired platform and geographic independent visual technology lends itself easily to deployment on the WWW. Many languages exist for development of tools on the web, including markup languages such as hypertext markup language (HTML), and scripting languages such as Perl, Active Server Pages (ASP), and hypertext preprocessor (PHP), which are particularly appropriate [75–78]. In the context of the VDSM, the PHP scripting language is incorporated with the HTML in the development of a user interface. For the visual display of the system topology model, the virtual reality modeling language (VRML) is used [79,80]. This modeling language is easily conveyed over the WWW using text files, and can be viewed using readily available browser plug-ins available for most operating systems. For data storage, the primary data set uses a relational database to track the potentially large and cumbersome number of metrics displayed, with some very small text-based files used to store settings and smaller pieces of data.

V. Implementation of VDSM

The VDSM uses the previously described visual cues to enable a designer to intuitively interact with a complex system topology. Figure 4 displays the developed web-based interface. In the upper left hand corner is the control interface for the system. Here, a designer can select the option to view the DSM, the numerical description of the subsystems, and all of the sensitivity information. The lower left corner of the web-based interface contains the introduced error monitoring function. As couplings are suspended, the introduced error levels will change in this window, alerting the designer if they exceed one or more of their preset error limits through a display of the introduced error for each constraint. The main section of the web-based interface is the display. This large area is where the module description data, the sensitivity data, and the VDSM is displayed. All of the designer interaction takes place within this main window.

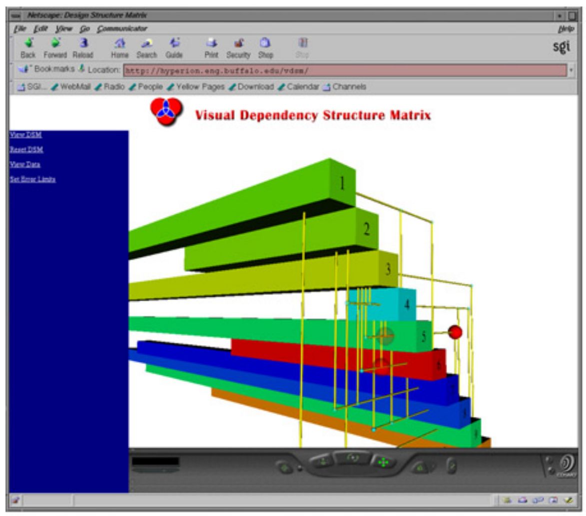
A. Test Problem One

A sample 10 subsystem problem is displayed in the main window of the system in Fig. 4. Each module's identification is printed on its face. The time for each analysis module is represented through color variation, similar to a stress strain plot (with the smallest time modules being blue and the largest red). Cost is represented by a depth cue, and can easily be seen in Figs. 5a and 5b, in which rotated views are shown. The displayed module depth is normalized using the minimum cost as a baseline. The module that has the largest depth has the highest cost to execute. Looking at these figures, it is evident that modules 7 and 8 both have very low execution times (blue), but extremely high costs (very deep). Module 6 has a large execution time and a moderate cost.

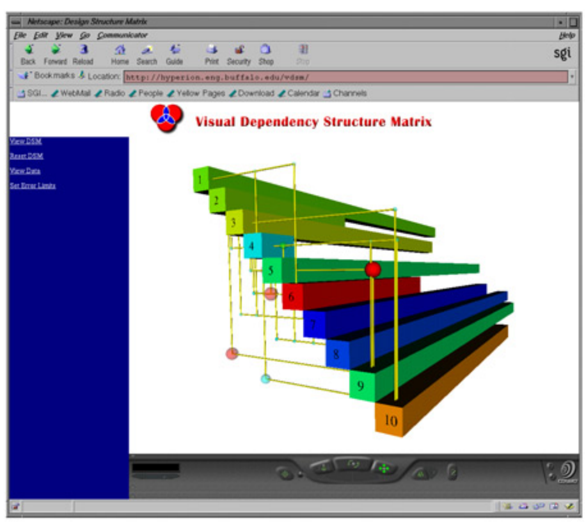
The linkage between subsystems is shown through the pipes and nodes that are superficially similar to the two-dimensional DSM. The difference here is that depth has been added to clarify which subsystem outputs feed which subsystem inputs. In this manner, the confusion that can result from lines crossing each other in Fig. 4 has been eliminated. This improvement can be seen clearly in Figs. 6a and 6b, which show the linkages between several different subsystems all crossing in a close vicinity. From these figures, it is easy here to track where the output from subsystem 5 flows. In the two-dimensional DSM, the crossovers might have caused confusion.

Recall that the coupling impact is visualized using the round nodes to indicate the coupling's characteristics. The impact of the coupling on the downstream analysis module's output is indicated using the visual cue of transparency. If a subsystem's output has little impact on another subsystem's output, the node is almost completely transparent. Fig. 6b shows that the link between modules 5 and 4 is very weak, given the almost transparent large red coupling. Recall that the color variation scheme is also used in displaying coupling information to indicate the degree of average error introduced into the constraints by suspending the coupling. The red color of this coupling (that is, between 5 and 4) means that if that coupling were suspended, a considerable amount of error would be introduced into the constraints. This suggests to the designer that the coupling should not be suspended. The final visual cue used by the VDSM to convey coupling information is the size of the spheres, which corresponds to degree of error introduced into the objective function as a result of suspending that coupling. The size of the sphere between module 5 and 4 is rather large, indicating that a significant amount of error would be introduced into the objective function. Hence, to summarize, the designer would immediately understand that this coupling, while very weak in the local sense, has a strong impact in the overall optimization problem and should not, therefore, be suspended.

Given the VDSM representation of Fig. 4, with the ability to rotate and zoom in on any specific area (as in Figs. 5 and 6), the designer can now use all of these visual cues to select which couplings to suspend or remove for the analysis phase of the optimization problem. It should be noted that the designer may also elect to run an external tool to resequence the analysis modules so as to result in a quicker analysis convergence.



a) View 1

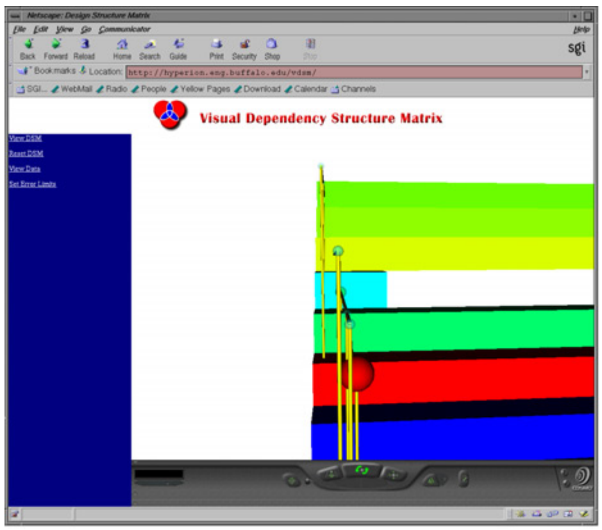


b) View 2

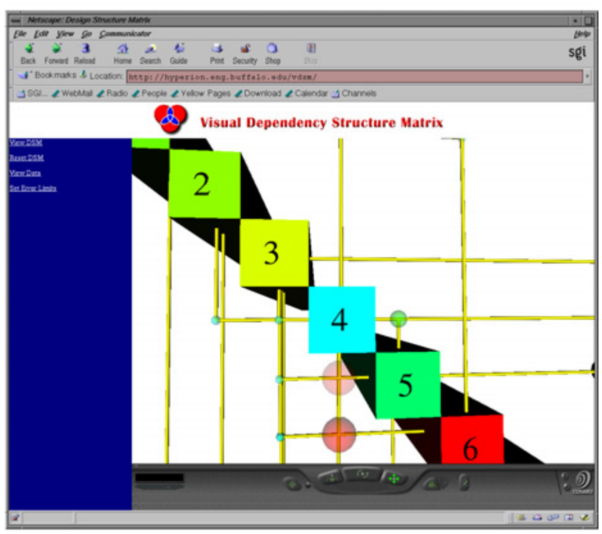
Fig. 5 Rotated VDSM.

Figure 7a shows the front view representation of the VDSM once a coupling has been chosen for suspension. Fig. 7b shows the error estimates and limits associated with that particular coupling suspension decision. Fig. 7a shows that, despite the fact that the coupling between modules 5 and 9 is opaque, large and red (indicating strong local coupling with significant impact on both the objective function and the constraints), it is selected for suspension. Once the coupling is selected through the click of the mouse, the sphere color shifts to gray. The large red cues notify the designer that introduced error exceeds the chosen limits. This does not exclude the coupling from suspension, but does allow the designer to know that, according to the calculated impact (Fig. 7b), the introduced error would be too great to suspend that coupling.

The designer can continue to click on couplings, thereby selecting and deselecting them, and, in real-time, and receive feedback on the introduced error in the window at the left of the model. Figures 8a and 8b show the impact on error for two additional sets of couplings that are considered for suspension. In Fig. 8a, multiple couplings



a) View 1



b) View 2

Fig. 6 Couplings in rotated VDSM.

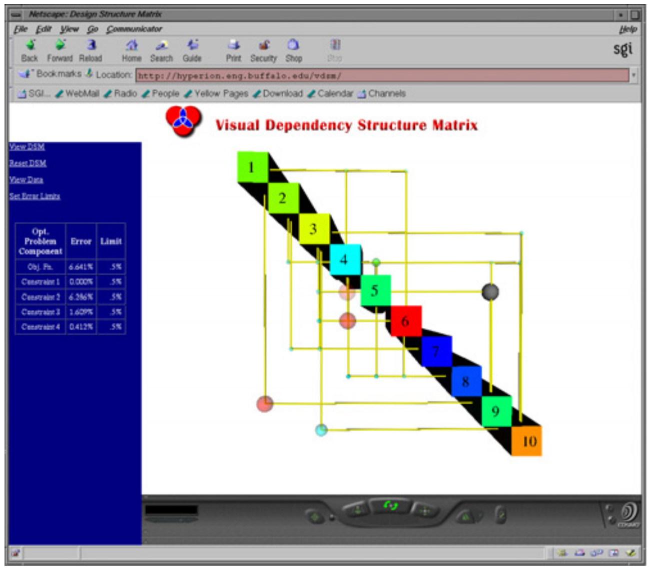
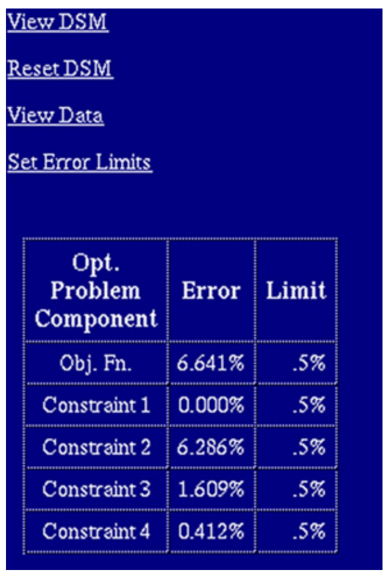


Fig. 7a VDSM with suspended coupling.



View DSM

Reset DSM

View Data

Set Error Limits

Opt. Problem Component	Error	Limit
Obj. Fn.	6.641%	.5%
Constraint 1	0.000%	.5%
Constraint 2	6.286%	.5%
Constraint 3	1.609%	.5%
Constraint 4	0.412%	.5%

Fig. 7b Estimated introduced error.

are selected, including those between modules 10 and 3 (not expected to have a particularly significant impact on objective function or constraints, given the smaller, green coupling), and modules 4 and 5 (an even smaller green coupling). In Fig. 8a, the objective function and constraint 2 errors now exceed the recommended limits, so different choices must be explored.

In Fig. 8b, the designer has found a final feasible set of couplings to suspend. Fifteen couplings are suspended, including ten feedbacks and five feedforwards. The only couplings remaining, in fact, are those from 9 to 1, 6 to 4, 5 to 4, 4 to 5, and 5 to 9. The convergence requirements for this 'reduced' system will now be substantially less than the full system. In practice, this task took approximately 1 to 2 min for this 10 subsystem problem. This can be compared with the large quantity of data to be tracked in the simple 5 subsystem example problem from Tables 1

Opt. Problem Component	Error	Limit
Obj. Fn.	0.924%	.5%
Constraint 1	0.000%	.5%
Constraint 2	0.875%	.5%
Constraint 3	0.224%	.5%
Constraint 4	0.057%	.5%

Fig. 8a Initial trial for suspensions.

Opt. Problem Component	Error	Limit
Obj. Fn.	0.004%	.5%
Constraint 1	0.000%	.5%
Constraint 2	0.004%	.5%
Constraint 3	0.001%	.5%
Constraint 4	0.000%	.5%

Fig. 8b Final choice of 15 suspended couplings.

to 3. This simple series of investigations demonstrate that this tool allows a designer intuitively to interact with the system topology in real-time to achieve a tradeoff capability. Although a formal heuristic optimization approach may find a solution more quickly for a problem of this dimensionality, this tool provides the means for designers to make informed decisions based on all the same data.

The 10 module example shown in Figs. 4–8 demonstrate that the first couplings to select are the small, blue and green, faint ones, which correspond to the least important couplings in both a local and global sense. If these do not exceed the prescribed error limits, one would then work up through the color and size spectrum, investigating the next best coupling that could be suspended. The designer should certainly concentrate on feedbacks, rather than feedforwards, given that the feedbacks are those couplings that represent iteration. However, even suspending feedforwards results in cost savings, as those derivatives need not be calculated in that optimization cycle.

Figure 9 shows the reduced system. In Fig. 9, there will still be an iterative loop enclosing most all the modules involved (due to the retention of the 9 to 1 coupling). The retention of couplings 5 to 4, and 6 to 4 encompass two modules (5 and 6) for which one is very expensive to implement but doesn't take much time (5) while the other is of moderate expense but takes a substantial amount of time. Coupling 4 to 5 was also retained, but module 4 is neither expensive nor time-consuming to implement. Lastly, the coupling from 5 to 9 is a feedforward, which will not require iteration, so the cost of modules associated with this link are not of significant interest in the tradeoff study. All of this information, easily gained by the swivel of the VDSM image, can suggest to the designer that a fixed point iteration scheme might not be the best way to converge this reduced system. Rather, an approach involving an additional interior iteration might actually speed the overall convergence of the system, enabling those modules with the greatest coupling (4 through 6) to converge faster, which would then propagate to a faster overall system convergence. An investment of additional cost and time for reimplementation of these modules could save one or more system level iterations.

Figure 10 shows the tradeoffs between the final accuracy achieved for the objective function (normalized with respect to the objective function obtained with no suspensions) and the computational savings achieved by implementing suspensions using VDSM (again, normalized with respect to the original time taken for the unsuspended problem). Recall that the true value of VDSM is as a planning tool before a formal optimization. VDSM will enable

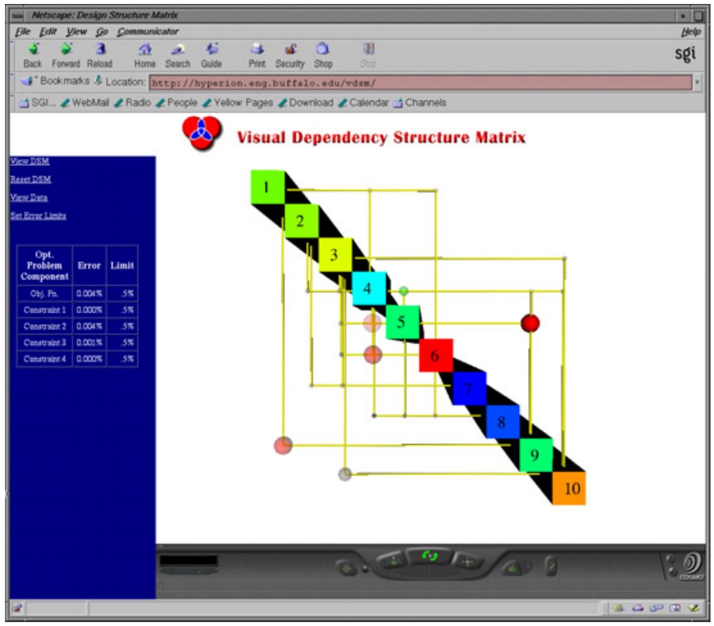


Fig. 9 Reduced system following coupling suspension.

the designer to structure the topology of the participating analyses to rapidly achieve an approximate solution, which provides a starting point for a formal optimization process if desired. In Fig. 10, at very low allowable errors for the objective function, there is a negligible impact on objective function accuracy but an increase in resulting computational costs. For low allowable errors, the set of suspended couplings will rapidly change. Hence, in the final convergence process of the optimization problem, this results in a lack of critical information when it is needed most. In contrast, at very large allowable errors, the suspended coupling set is relatively stable, producing a more rapid and stable convergence process. However, at the very high end of the allowable error limit, the final objective function values that vary widely and are, on the whole, not sufficiently desirable. In the middle of these two extremes, from allowable error limits of approximately 0.5% to 25%, the suspended objective function values have only about 20% difference from the original unsuspended case, while achieving significant computational savings. While the desirable ranges of the allowable error limits (that is, to obtain the least accuracy losses with greatest computational savings) will be problem dependent, the designer can rapidly explore these limits interactively using the VDSM interface. Further, the computational savings achieved when using the suspension approach will enable the designer to explore more alternatives in the same amount of time.

B. Test Problem Two

Figure 11a shows the VDSM for a much larger problem while Fig. 11b shows the traditional flow chart representation. In this case, the system level problem is an aircraft design problem. Consider the confusion of the web of arrows and blocks in Fig. 11b. The only information conveyed in this diagram is the connectivity. No information is provided about module costs or times and it is extremely difficult to determine order of task (i.e. module) implementation. While the local and total sensitivity information is not available for this example, the VDSM still provides a useful means for a designer to gain and share insight related to a system analysis topology. It can be seen from Fig. **??** that modules 3, 4, and 5 are all long duration modules, with relatively low execution cost. Without resequencing or error introduction studies, a designer could use the VDSM in this case to target these three modules for platform dependent decisions. For example, these analyses could be assigned to higher performance machines. Alternatively, it might be determined to use parallel processing so that these modules would not slow down the rest of the subsystem analyses.

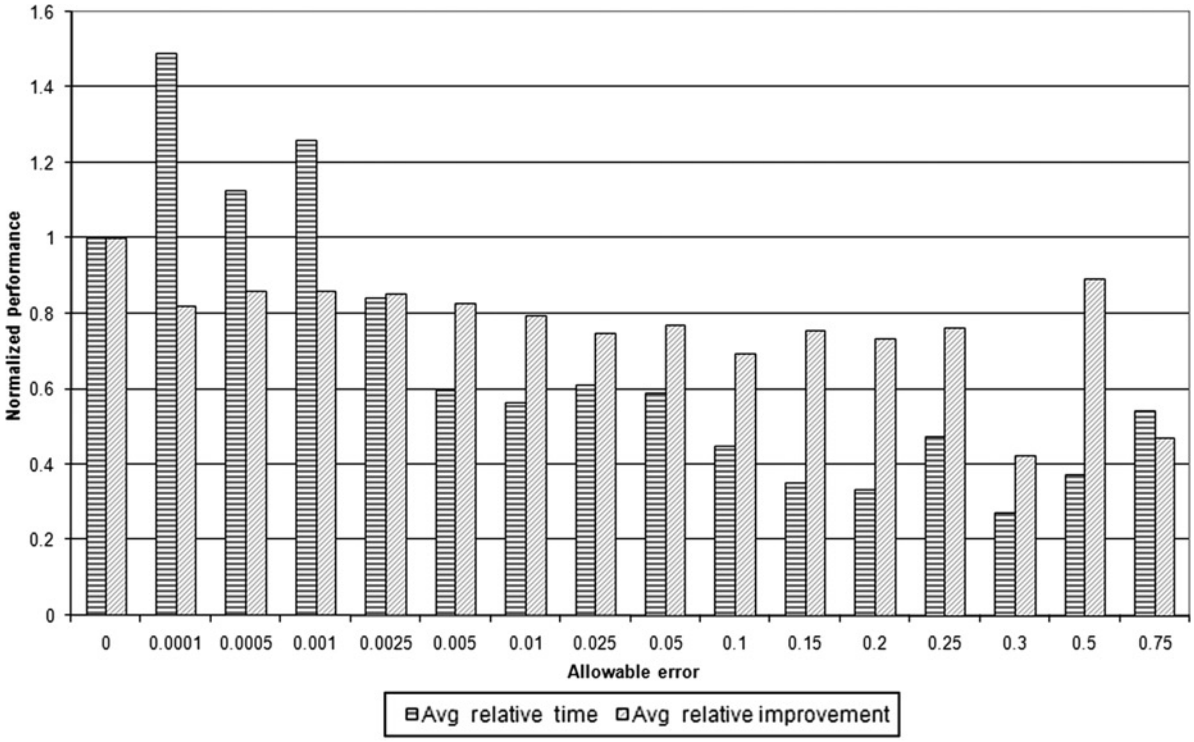


Fig. 10 Tradeoff between accuracy and efficiency

Additionally, modules 6, 7, and 8 all have a low execution time, but higher cost. If these were PC-based analyses, the manager might wish to trade the resources used for these operations with those used for modules 3 to 5, leveling the resource allocation. This benefit is completely unrelated to the use of the VDSM as a coupling suspension tool, but points out the usefulness and flexibility derived from viewing system topology data visually, instead of through a set of spreadsheets.

In some cases, a designer may wish to view the actual data, rather than the visual cues representing that data. The specific coupling sensitivity and module information is also available through the VDSM tool, either through selecting an object of interest, or following a link to the data store. This allows for troubleshooting of questionable representations, as well as very specific manipulation of couplings for suspension.

After viewing the aircraft design example, it becomes clear that there is an upper bound to the usefulness of the VDSM in allowing designers to understand their system topology at a glance. To go beyond a 20 subsystem sized system, which is moderately coupled, using a standard size display, designers may wish to use nesting technique to group modules together (that is, show by discipline or functional group rather than by specific analysis module). An alternative solution to the display of systems with large numbers of modules would be the use of large-scale display tools, such as a Powerwall™. The real limiting factor in using the VDSM, beyond the actual visualization of the modules themselves, is the number of couplings. Even with the intuitive interface, very large systems may prove cumbersome if nesting-type strategies are not used.

There is still a need for a computer-based surrogate selection capacity for problems that are too large, or analysis times that are too short for application of the VDSM. Several strategies have been developed and implemented for solving the complex binary coupling selection problem [22,29,34,56,58,81]. This surrogate capability becomes important both in the case of unmanageably complex systems, as well as when designers wish to allow the optimization to proceed unattended. A requirement for allowing a computer to serve as a capable surrogate for a designer's intuition and experience, is that an efficient and robust technique must be used to select couplings for suspension.

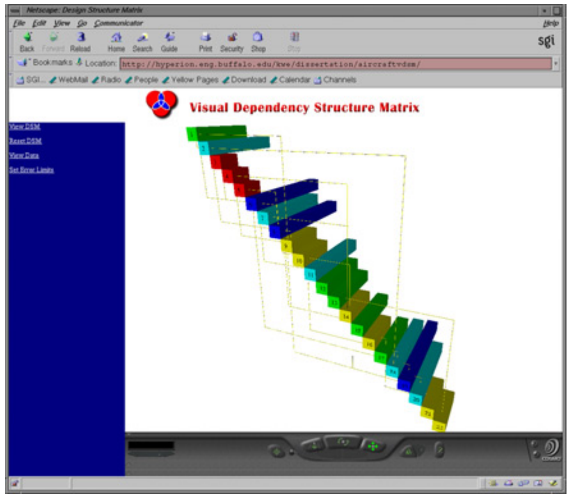


Fig. 11a VDSM for larger aircraft design.

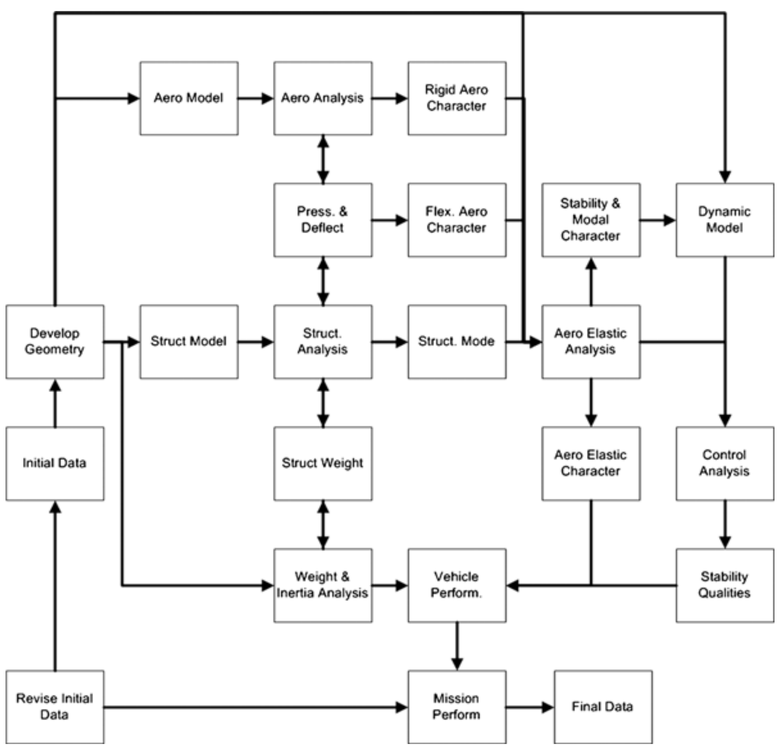


Fig. 11b Traditional flowchart.

VI. Conclusions

In this paper, the need for an intuitive methodology for designers to interact with their system analysis topologies was demonstrated. Previous efforts in this area had been confined to two dimensional representations, with their basis in graph theory. Through the development of the VDSM, a completely innovative approach has been developed. The VDSM uses visual cues to intuitively and interactively convey system analysis semantics to a designer. Designers now have the capability to visually process such issues as analysis module time and cost, local coupling sensitivity, and total derivative-based sensitivity metrics. This method now overcomes the obstacles that paper-based tradeoff activities typically encounter.

A surprising benefit of the VDSM has been demonstrated in the sample case of the aircraft design problem. Even without coupling sensitivity information, considerable insight can be gained into the topology of such a complex analysis. Analyses that may require additional computational resources can be identified for better resource allocation. The potential for application is only limited by how designers choose to use this technique. The use of a web-based architecture allows for platform independent visualization and sharing of information. Web-based tools are easily expanded and integrated, allowing for the vast potential for synergistic effects if this work is combined with other visual design steering tools.

However, the VDSM methodology is not a universal solution. Even using this intuitive methodology, the number of modules or couplings may grow too large to effectively interact with. In the case of systems with greater than approximately 20 modules, standard display screens prove limiting. This dissertation proposes two alternative schemes to address this issue, allowing a designer to continue to employ the VDSM methodology. The first strategy is to find a larger display. Designers could view the VDSM on a larger screen such as is available with Immersadesks^{TM} or Powerwalls^{TM}. The second technique is to reduce the quantity of information to be displayed by using a nesting strategy.

Acknowledgments

The authors wish to acknowledge partial support of this work by NYSTAR and NSF Grant DDMI 9553210.

References

[1] Bloebaum, C. L., Hajela, P., and Sobieszczanski-Sobieski, J., "Decomposition Methods for Multidisciplinary Synthesis," *Design and Optimization Techniques and their Application, Controls and Dynamic Systems*, Vol. 57, Academic International Press, New York/London/Orlando, FL, 1993, pp. 1–24.

[2] Sobieszczanski-Sobieski, J., "Multidisciplinary Design Optimization: An Emerging, New Engineering Discipline," *Advances in Structural Optimization*, Kluwer Academic, Norwell, MA, 1995, pp. 483–496.

[3] Sobieszczanski-Sobieski, J., and Haftka, R., "Multidisciplinary Aerospace Design Optimization: Survey of Recent Developments," *Structural and Multidisciplinary Optimization*, Vol. 14, No. 1, 1997, pp. 1–23.

[4] Kodiyalam, S., and Sobieszczanski-Sobieski, J., "Multidisciplinary Design Optimization: Some Formal Methods, Framework Requirements, and Application to Vehicle Design," *International Journal of Vehicle Design*, Vol. 25, No. 1–2, 2001, pp. 3–22.

[5] Kodiyalam, S., Yang, R. J., Gu, L., and Tho, C.-H., "Multidisciplinary Design Optimization of a Vehicle System in a Scalable, High Performance Computing Environment," *Structural and Multidisciplinary Optimization*, Vol. 26, No. 3–4, 2004, pp. 256–263.
doi: 10.1007/s00158-003-0343-2

[6] Sobieski, J., "A Linear Decomposition Method for Optimization Problems: Blueprint for Development," NASA Technical Memorandum 83248, 1982.

[7] Sobieszczanski-Sobieski, J., "Optimization by Decomposition: A Step from Hierarchic to Non-Hierarchic Systems," Recent Advances in Multidisciplinary Analysis and Optimization, NASA CP 3031, 1988.

[8] Winer, E. H., and Bloebaum, C. L., "Visual Design Steering for Optimization Solution Improvement," *Structural and Multidisciplinary Optimization,* Vol. 22, No. 3, 2001, pp. 219–229.
doi: 10.1007/s001580100139

[9] Winer, E. H., and Bloebaum, C. L., "Development of Visual Design Steering as an Aid in Large Scale Multidisciplinary Design Optimization. Part I: Method Development," *Structural and Multidisciplinary Optimization,* Vol. 23, No. 6, July 2002, pp. 412–424.
doi: 10.1007/s00158-002-0203-5

[10] Winer, E. H., and Bloebaum, C. L., "Development of Visual Design Steering as an Aid in Large Scale Multidisciplinary Design Optimization. Part II: Method Validation," *Structural and Multidisciplinary Optimization,* Vol. 23, No. 6, July 2002, pp. 425–435.
doi: 10.1007/s00158-002-0204-4

[11] Hajela, P., Bloebaum, C. L., and Sobieszczanski-Sobieski, J., "Application of Global Sensitivity Equations in Multidisciplinary Aircraft Synthesis," *Journal of Aircraft*, Vol. 27, No. 12, 1990, pp. 1002–1010.
doi: 10.2514/3.45974

[12] Sobieszczanski-Sobieski, J., "On the Sensitivity of Complex, Internally Coupled Systems," *AIAA Journal*, Vol. 28, No. 1, 1990, pp. 153–160.
doi: 10.2514/3.10366
[13] Sobieszczanski-Sobieski, J., "High Order Sensitivity Analysis of Complex, Coupled Systems," *AIAA Journal*, Vol. 28, No. 4, 1990, pp. 756–758.
doi: 10.2514/3.10458
[14] Sobieszczanski-Sobieski, J., Bloebaum, C. L., and Hajela, P., "Sensitivity of Control-Augmented Structure Obtained by a System Decomposition Method," *AIAA Journal*, Vol. 29, No. 2, 1991, pp. 264–270.
doi: 10.2514/3.10573
[15] Bloebaum, C. L., Hajela, P., and Sobieszczanski-Sobieski, J., "Non-Hierarchic System Decomposition in Structural Optimization," *3rd USAF/NASA Symposium on Recent Advances in Multidisciplinary Analysis and Optimization*, San Francisco, CA, Sept. 1990.
[16] Bloebaum, C. L., "Formal and Heuristic System Decomposition Methods in Multidisciplinary Synthesis," NASA Contractor Report, CR-4413, Hampton, VA, Dec.1991.
[17] Bloebaum, C. L., "Formal and Heuristic System Decomposition Methods in Multidisciplinary Synthesis," Ph.D. Thesis, University of Florida, Gainsville, FL, 1991.
[18] Bloebaum, C. L., Hajela, P., and Sobieszczanski-Sobieski, J., "Non-Hierachic System Decomposition in Structural Optimization," *Engineering Optimization*, Vol. 19, No. 3, 1992, pp. 171–186.
doi: 10.1080/03052159208941227
[19] Cramer, E. J., Dennis, J. E., Frank, P. D., Lewis, R. M., and Shubin, G. R., "Problem Formulation for Multidisciplinary Optimization," *SIAM Journal of Optimization*, No. 4, 1994, pp. 754–776.
doi: 10.1137/0804044
[20] Balling, R. J., and Sobieszczanski-Sobieski, J., "Optimization of Coupled Systems: A Critical Overview of Approaches," *AIAA Journal*, Vol. 34, No. 1, 1996, pp. 6–17.
doi: 10.2514/3.13015
[21] Balling, R. J., and Wilkinson, C. A., "Execution of Multidisciplinary Design Optimization Approaches on Common Test Problems," *AIAA Journal*, Vol. 35, No. 1, 1997, pp. 178–186.
[22] Bloebaum, C. L., "Coupling Strength-based System Reduction for Complex Engineering Design," *Structural Optimization*, Vol. 10, No. 2, 1995, pp. 113–121.
doi: 10.1007/BF01743538
[23] English, K., Bloebaum, C. L., and Miller, E., "Development of Multiple Cycle Coupling Suspension in the Optimization of Complex Systems," *Structural Optimization*, Vol. 22, No. 4, 2001, pp. 268–283.
[24] Rogers, J. L., and Bloebaum, C. L., "Ordering Design Tasks Based on Coupling Strengths," *5th AIAA/USAF/NASA/OAI Symposium on Multidisciplinary Analysis and Optimization*, American Institute of Aeronautics and Astronautics (AIAA), Panama City, FL, September 1994.
[25] Miller, E., "Coupling Suspension and Elimination in Multidisciplinary Design Optimization," M.S. Thesis, State University of New York at Buffalo, Buffalo, NY, 1995.
[26] Aho, A. V., Hopcroft, J. E., and Ullman, J. D., *Data Structures and Algorithms*, Addison-Wesley Longman, Reading, MA, 1983, pp.198–229.
[27] Kerzner, H., *Project Management*, Reinhold, New York, NY, 1989.
[28] Neumann, K., and Steinhardt, U., *GERT Networks and the Time-Oriented Evaluation of Projects*, Springer-Verlag, Berlin/New York/Heidelberg, 1979.
[29] Steward, D. V., *System Analysis and Management*, Petrocelli Books, New York, NY, 1981.
[30] Rogers, J. L., "Knowledge-Based Tool for Decomposing Complex Design Problems," *ASCE Journal of Computing in Civil Engineering*, Vol. 4, No. 4, 1990, pp. 298–312.
doi: 10.1061/(ASCE)0887-3801(1990)4:4(298)
[31] McCulley, C., and Bloebaum, C. L., "Optimal Sequencing for Complex Engineering Systems Using Genetic Algorithms," *Fifth AIAA/USAF/NASA/OAI Symposium on Multidisciplinary Analysis and Optimization*, Panama City Beach, FL, September 1994, AIAA, pp. 718–730.
[32] McCulley, C., and Bloebaum, C. L., "A Genetic Tool for Optimal Design Sequencing in Complex Engineering Systems," *Structural Optimization*, Vol. 12, No. 2/3, 1996, pp. 186–201.
doi: 10.1007/BF01196956
[33] Rogers, J. L., McCulley, C. M., and Bloebaum, C. L., "Optimizing the Process Flow of Complex Design Projects," *Design Optimization, International Journal for Product and Process Improvement*, Vol. 1, No. 3, 1999, pp. 281–292.

[34] Eppinger, S. D., Whitney, D. E., Smith, R. P., and Gebala, D. A., "Organizing the Tasks in Complex Design Projects," *ASME Conference on Design Theory and Methodology*, Chicago, IL, September 1990, pp. 39–46.

[35] Rogers, J. L., "DeMAID: A Design Manager's Aide for Intelligent Decomposition Users Guide," NASA Technical Memorandum 101575, March 1989.

[36] Rogers, J. L., A Knowledge-Based Tool for Multilevel Decomposition of a Complex Design Problem, NASA Technical Paper 2903, May, 1989.

[37] Krishnan, V., Eppinger, S. D., and Whitney, D. E., "Towards A Cooperative Design Methodology: Analysis of Sequential Design Strategies," *ASME Conference on Design Theory and Methodology*, New York, NY, September 1991, pp. 165–172.

[38] Krishnan, V., Eppinger, S. D., and Whitney, D. E., "A Model-based Framework to Overlap Product Development Activities," *Management Science*, Vol. 43, No. 4, 1997, pp. 437–451.

[39] Krishnan, V., Eppinger, S. D., and Whitney, D. E., "Simplifying Iterations in Cross-Functional Design Decision Making," *Journal of Mechanical Design*, Vol. 119, No. 4, 1997, pp. 485–493.
doi: 10.1115/1.2826394

[40] Pimmler, T. P., and Eppinger, S. D., "Integration Analysis of Product Decompositions," *ASME Conference on Design Theory and Methodology*, Minneapolis, MN, September 1994, pp. 343–351.

[41] Kusiak, A., and Wang, J., "Decomposition in Concurrent Design," *Concurrent Engineering: Automation, Tools, and Techniques*, edited by A. Kusiak, Wiley, New York, NY, 1993, pp. 481–507.

[42] Kusiak, A., and Wang, J., "Efficient Organizing of Design Activities," *International Journal of Production Research*, Vol. 31, No. 4, 1993, pp. 753–769.

[43] Kusiak, A., and Wang, J., "Decomposition of the Design Process," *Journal of Mechanical Design*, Vol. 115, No. 4, 1993, pp. 687–695.
doi: 10.1115/1.2919255

[44] Kusiak, A., Larson, T. N., and Wang, J., "Reengineering of Design and Manufacturing Processes," *Computer Industry Engineering*, Vol. 26, No. 3, 1994, pp. 521–536.
doi: 10.1016/0360-8352(94)90048-5

[45] Kusiak, A., Larson, T. N., and Wang, J., "Reengineering of Design and Manufacturing Processes," *Computer Industry Engineering*, Vol. 26, No. 3, 1994, pp. 521–536.
doi: 10.1016/0360-8352(94)90048-5

[46] Kusiak, A., Wang, J., He, D. W., and Feng, C., "A Structured Approach for Analysis of Design Processes," *IEEE Transactions on Components, Packaging, and Manufacturing Technology*, Part A, Vol. 18, No. 3, 1995, pp. 664–673.
doi: 10.1109/95.465167

[47] Kusiak, A., and Larson, T. N., "Decomposition and Representation Methods in Mechanical Design," *Journal of Mechanical Design*, Vol. 117, No. 3, 1995, pp. 17–24.

[48] Kusiak, A., Wang, J., He, D. W., and Feng, C., "A Structured Approach for Analysis of Design Processes," *IEEE Transactions on Components, Packaging, and Manufacturing Technology*, Part A, Vol. 18, No. 3, 1995, pp. 664–673.
doi: 10.1109/95.465167

[49] Kusiak, A., "Integrated Product and Process Design: A Modularity Perspective," *Journal of Engineering Design*, Vol. 13, No. 3, 2002, pp. 223–231.
doi: 10.1080/09544820110108926

[50] Guo, W., Cha, J., Fang, Y., and Woo, T. C., "Concurrent Design: An Algebraic Perspective," *International Journal of Information Technology*, Vol. 1, No. 1, 1995, pp. 17–32.

[51] Eppinger, S. D., "Model-Based Approaches to Managing Concurrent Engineering," *Journal of Engineering Design*, Vol. 2, No. 4, 1991, pp. 283–290.
doi: 10.1080/09544829108901686

[52] Eppinger, S. D., Whitney, D. E., Smith, R. P., and Gebala, D. A., "A Model-Based Method for Organizing Tasks in Product Development," *Research in Engineering Design*, Vol. 6, No. 1, 1994, pp. 1–13.
doi: 10.1007/BF01588087

[53] Eppinger, S. D., Nukala, M. V., and Whitney, D. E., "Generalized Models of Design Iteration Using Signal Flow Graphs," *Research In Engineering Design*, Vol. 9, No. 2, 1997, pp. 112–123.
doi: 10.1007/BF01596486

[54] Gebala, D. A., and Eppinger, S. D., "Methods for Analyzing Design Procedures," *ASME Conference on Design Theory and Methodology*, New York, NY, September 1991, pp. 227–233.

[55] Cho, S.-H., and, Eppinger, S. D., "Product Development Process Modeling Using Advanced Simulation," *ASME Conference on Design Theory and Methodology*, American Society of Mechanical Engineers, Fairfield, NJ, Pittsburgh, PA, September 2001.

[56] Braha, D., "Partitioning Tasks to Product Development Teams," *Proceedings of DETC'02: ASME 2002 Design Engineering Technical Conferences and Computers and Information in Engineering Conference*, American Society of Mechanical Engineers (ASME), Montreal, Canada, Sept. 29–Oct. 2, 2002.
[57] Browning, T. R., Deyst, J. J., Eppinger, S. D., and Whitney, D. E., "Adding Value in Product Development by Creating Information and Reducing Risk", *IEEE Transactions on Engineering Management*, Vol. 49, No. 4, 2002, pp. 443–458.
[58] Browning, T. R., "Applying the Design Structure Matrix to System Decomposition and Integration Problems: A Review and New Directions," *IEEE Transactions on Engineering Management*, Vol. 48, No. 3, 2001, pp. 292–306. doi: 10.1109/17.946528
[59] Browning, T. R., and Eppinger, S. D., "Modeling Impacts of Process Architecture on Cost and Schedule Risk in Product Development," *IEEE Transactions on Engineering Management*, Vol. 49, No. 4, 2002, pp. 428–442. doi: 10.1109/TEM.2002.806709
[60] Browning, T. R., Deyst, J. J., Whitney, D. E., and Eppinger, S. D., "Adding Value in Product Development by Creating Information and Reducing Risk," *IEEE Transactions on Engineering Management*, Vol. 49, No. 4, 2002, pp. 443–458. doi: 10.1109/TEM.2002.806710
[61] Sosa, M. E., Eppinger, S. D., and Rowles, C. M., "Identifying Modular and Integrative Systems and Their Impact on Design Team Interactions," *ASME Journal of Mechanical Design*, Vol. 125, June 2003, pp. 240–252. doi: 10.1115/1.1564074
[62] Smith, R. P., and Eppinger, S. D., "Identifying controlling Features of Engineering Design Iteration," *Management Science*, Vol. 43, No. 3, 1997, pp. 276–293.
[63] Rogers, J. L., "Tools and Techniques for Decomposing and Managing Complex Design Projects," *AIAA Journal of Aircraft*, Vol. 36, No. 1, 1999, pp. 266–274.
[64] Steward, D. V., "Re-Engineering the Design Process," *Proceedings of the 2nd Workshop on Enabling Technologies: Infrastructure for Collaborative Enterprises*, Morgantown, WV, 1993.
[65] Smith, R. P., and Eppinger, S. D., "A Predictive Model of Sequential Iteration in Engineering Design," *Management Science*, Vol. 43, No. 8, 1997, pp. 1104–1120.
[66] Rogers, J. L., "DeMAID/GA: An Enhanced Design Manager's Aid for Intelligent Decomposition," *6th AIAA/NASA/ISSMO Symposium on Multidisciplinary Analysis and Optimization*, American Institute of Aeronautics and Astronautics (AIAA), pp. 1497–1504.
[67] Michelena, N. F., and Papalambros, P. Y., "Optimal Model–Based Decomposition of Powertrain System Design," *Journal of Mechanical Design*, Vol. 117, No. 4, 1995, pp. 499–505. doi: 10.1115/1.2826710
[68] Liu, J., and Brown, D. C., "Generating Design Decomposition Knowledge for Parametric Design Problems," *Proceedings of the Artificial Intelligence in Design '94 Conference*, edited by J. S. Gero and F. Sudweeks, Kluwer Academic, Norwell, MA, 1994, pp. 661–678.
[69] Kroo, I. M., Altus, S. S., Braun, R. D., Gage, P. J., and Sobieski, I., "Multidisciplinary Optimization Methods for Aircraft Preliminary Design," *Fifth AIAA/USAF/NASA/OAI Symposium on Multidisciplinary Analysis and Optimization*, Panama City Beach, FL, September 1994, AIAA, pp. 697–707.
[70] Austin, S. A., Baldwin, A. N., and Steele, J. L., "Improving Building Design through Integrated Planning and Control," *Engineering Construction and Architectural Management*, Vol. 9, No. 3, 2002, p. 249.
[71] Stebbings, C., and Cooper, S.,"PlanWeaver," *7th Design Structure Matrix (DSM) Workshop*, Seattle, WA, October 2005.
[72] Danilovic, M.,"Tools for Implementing Information Driven Management Approach- Multiplan Professional and Complex Problem Solver," *7th Design Structure Matrix (DSM) Workshop*, Seattle, WA, October 2005.
[73] Hamilton, A., and Pallaver, M., "Acclaro Designer DSM software," *7th Design Structure Matrix (DSM) Workshop*, Seattle, WA, October 2005.
[74] Waldman, F., and Sangal, N., "Results of DSM Analysis of Software Systems in Industrial Applications," *7th Design Structure Matrix (DSM) Workshop*, Seattle, WA, October 2005.
[75] W3C, World Wide Web Consortium, HTML 4.01 Specification, http://www.w3.org/TR/html4/ [retrieved 23 July 2008].
[76] Fedorov, A., Francis, B., Maharry, D., Sussman, D., and Ullman, C., *ASP2.0: Programmer's Reference*, Wrox Press, 1998.
[77] Till, D., *Teach Yourself Perl5 in 21 Days*, Sams Publishing, Indianapolis, IN, 1996.
[78] PHP: Hypertext Preprocessor, http://www.php.net, 2000 [retrieved 23 July 2008].
[79] Ames, A. L., Nadeau, D. R., and Moreland, J. L., *The VRML Sourcebook*, Wiley, New York, NY, 1996.

[80] Ames, A. L., Nadeau, D. R., and Moreland, J. L., *The VRML 2.0 Sourcebook*, 2^{nd} ed., Wiley, New York, NY, 1996.
[81] English, K., and Bloebaum, C. L., "A Comparison of Optimization Techniques for Solving the Coupling Suspension Problem," *Proceedings of 8th AIAA/NASA/ISSMO Symposium on Multidisciplinary Analysis and Optimization*, American Institute of Aeronautics and Astronautics (AIAA), Long Beach, CA, September 2000.

Paul Plassermann
Associate Editor