Proceedings of the ASME 2007 International Design Engineering Technical Conferences & Computers and Information in Engineering Conference
IDETC/CIE 2007
September 4-7, 2007, Las Vegas, Nevada, USA

DETC2007-34698

OPTIMAL PARTITIONING AND COORDINATION DECISIONS IN DECOMPOSITION-BASED DESIGN OPTIMIZATION

James T. Allison, Michael Kokkolaras, and Panos Y. Papalambros
Optimal Design Laboratory
Department of Mechanical Engineering
University of Michigan, Ann Arbor, Michigan 48109
Email: {optimize,mk,pyp}@umich.edu

ABSTRACT

Solution of complex system design problems using distributed, decomposition-based optimization methods requires determination of appropriate problem partitioning and coordination strategies. Previous optimal partitioning techniques have not addressed the coordination issue explicitly. This article presents a formal approach to simultaneous partitioning and coordination strategy decisions that can provide insights on whether a decomposition-based method will be effective for a given problem. Pareto-optimal solutions are generated to quantify tradeoffs between the sizes of subproblems and coordination problems, as measures of the computational costs resulting from different partitioning-coordination strategies. Promising preliminary results with small test problems are presented. The approach is illustrated on an electric water pump design problem.

1 Introduction

Numerous methods have been developed to solve complex system design problems partitioned into smaller subproblems. This decomposition-based approach can ease several difficulties encountered in system design, such as computational expense and management of complex interactions between system elements. Solving a problem in this way requires system designers to decide how to partition the system into subproblems and how to coordinate solution of subproblems toward a consistent, optimal system design. Partitioning decisions have been studied analytically, while only qualitative guidance exists in the literature for selecting an appropriate coordination method. The interaction between partitioning and coordination (P/C) decisions has not been studied systematically, but one expects that partitioning decisions will influence coordination decisions, and vice versa. In this article an optimal partitioning and coordination decision-making model is formulated and solved for test problems. Initial results indicate that accounting for the interaction between partitioning and coordination can lead to better decomposition strategies.

1.1 Decomposition-based System Design

The system design problems considered here involve multidisciplinary, coupled analyses (e.g., a set of coupled CAE simulations) where input/output properties are assumed to be known precisely. The vector of quantities computed by the j-th analysis function and required as input to the i-th analysis function is termed analysis coupling variable $\mathbf{y}_{ij}$. The vector of all coupling variables input to analysis i from any other analysis in the system is $\mathbf{y}_i$, and all design variables required as input to analysis i form the vector $\mathbf{x}_i$. In this manner, we define the i-th analysis function as $\mathbf{a}_i(\mathbf{x}_i, \mathbf{y}_i)$. Design variables that are inputs to $\mathbf{a}_i(\mathbf{x}_i, \mathbf{y}_i)$ only are termed local variables $\mathbf{x}_{\ell i}$; design variables that are inputs to $\mathbf{a}_i(\mathbf{x}_i, \mathbf{y}_i)$ and at least one other function are termed shared variables $\mathbf{x}_{si}$. Shared and local variables together form $\mathbf{x}_i = [\mathbf{x}_{\ell i}, \mathbf{x}_{si}]$ (vectors are assumed to be row vectors). The collections of all design variables, coupling variables, and analysis functions are $\mathbf{x}$, $\mathbf{y}$, and $\mathbf{a}(\mathbf{x}, \mathbf{y})$, respectively. Shared and coupling variables for subproblem i comprise its set of linking variables $\mathbf{z}_i$.

A system is consistent if the values of all copies of a shared variable agree for all shared variables, and if the value of every coupling variable is equal to its corresponding analysis output. More precisely, shared variable consistency is achieved if

$$x_q^{(k)} = x_q^{(l)} \quad \forall\, k \neq l, \quad k,l \in D_s(x_q) \tag{1}$$

is satisfied for all shared variables, where x_q is a component of $\mathbf{x}$ that is shared among the analysis functions $\mathbf{a}_i(\mathbf{x}_i, \mathbf{y}_i)\ \forall i \in D_s(x_q)$, with $D_s(x_q)$ being the set of indices of analysis functions that depend on the shared variable x_q; superscripts indicate the analysis function where the shared variable copy is input. Coupling variable consistency is achieved, if for every coupling variable

$$\mathbf{y}_{ij} - \mathbf{a}_j(\mathbf{x}_j, \mathbf{y}_j) = \mathbf{0} \tag{2}$$

is satisfied. The set of all such equality constraints is $\mathbf{y} - \mathbf{S}\mathbf{a}(\mathbf{x},\mathbf{y}) = \mathbf{0}$, where $\mathbf{S}$ is a boolean selection matrix that extracts the components of $\mathbf{a}(\mathbf{x},\mathbf{y})$ that correspond to the components of $\mathbf{y}$. These coupling variable consistency constraints are referred to as the system analysis equations. Equations (1) and (2) together form the system consistency constraints.

The optimal system design problem is formulated as

$$\begin{aligned} \min_{\mathbf{x}} \quad & f(\mathbf{x}, \mathbf{y}_\mathrm{p}(\mathbf{x})) \\ \text{subject to} \quad & \mathbf{g}(\mathbf{x}, \mathbf{y}_\mathrm{p}(\mathbf{x})) \leq \mathbf{0} \\ & \mathbf{h}(\mathbf{x}, \mathbf{y}_\mathrm{p}(\mathbf{x})) = \mathbf{0}, \end{aligned} \tag{3}$$

where $\mathbf{y}_p(\mathbf{x})$ is a solution to the system analysis equations for a given design, and the objective and constraint function values are outputs of a subset of analysis functions. This formulation is known as multidisciplinary feasible (MDF) [1] or All-in-One (AiO), and implicitly achieves shared variable consistency. For every optimization iterate $\mathbf{x}$ the system analysis equations must be solved for $\mathbf{y}_p(\mathbf{x})$.

If no feedback loops exist among analysis functions, the system analysis equations can be satisfied simply by executing the analysis functions in the proper sequence; analysis function outputs give the coupling variable values directly. An iterative algorithm is required for system analysis if feedback loops exist. Alternatively, the optimization algorithm can solve for $\mathbf{y}_p(\mathbf{x})$ using equality constraints to enforce coupling variable consistency. This enables coarse-grained parallel processing and can ease numerical difficulties associated with strongly coupled analysis functions [2]. The Individual Disciplinary Feasible (IDF) formulation is the simplest way to use this approach [1]. Balling and Sobieski suggested a hybrid approach that uses function sequencing to satisfy feedforward coupling relationships, and equality constraints to satisfy feedback coupling relationships [3].

Distributed system optimization formulations are applied to systems that have been partitioned into smaller subproblems. A separate optimization problem is defined for each subproblem; a coordination algorithm guides the repeated solution of subproblems toward a state of system consistency and optimality. Distributed optimization can employ equality constraints or penalty functions within subproblem optimization formulations to help satisfy system analysis equations. The set of design variables that are inputs for the functions in subproblem i and at least one other subproblem are the external shared variables $\bar{\mathbf{x}}_{si}$. Coupling variables passed from functions in subproblem j to subproblem i are the external coupling variables $\bar{\mathbf{y}}_{ij}$. Independent subproblem solution requires local copies not only of external coupling variables, but also of external shared variables. The coordination algorithm must ensure all copies match at convergence, satisfying the system consistency constraints. Some examples of coordination algorithms include optimization algorithms (e.g., Collaborative Optimization (CO) [4]), fixed point iteration (e.g., Analytical Target Cascading (ATC) [5]), Newton's method (e.g., ATC [6]), and penalty methods (e.g., ATC and Augmented Lagrangian Decomposition (ALD) [7]). Distributed methods are appropriate when systems are large and sparsely connected [8], when the design environment is distributed [9], or when specialized optimization algorithms can be exploited for solving particular subproblems [10, 11]. The techniques introduced in this article can be used to assess quantitatively whether distributed optimization is appropriate for a particular problem.

1.2 Partitioning and Coordination

The partitioning problem (P) is to decide which of m analysis functions should be clustered into each of the N subproblems. The coordination problem (C) is to specify a method for satisfying system consistency requirements; this typically consists of consistency constraint management and subproblem solution strategy. Graph theory can be used to study P/C decisions. Design variables and analysis functions are viewed as vertices in a directed graph whose arcs indicate a dependency relationship. For example, consider the following system of analysis functions:

$$\begin{aligned} & a_1(x_1, x_2, x_4) \\ & a_2(x_6, y_{21}) \\ & a_3(x_2, x_3, x_4, y_{31}, y_{34}) \\ & a_4(x_4, x_5, y_{41}). \end{aligned}$$

A directed graph representation of this system is shown in Fig. 1. If a system's graph is acyclic, then no feedback couplings exist.

A system graph can be represented compactly using its adjacency matrix. Since the n design variables are independent, the lower n rows of the adjacency matrix are empty and may be omitted without loss of information. The reduced system adjacency matrix for the above example is:

$$\mathbf{A} = \begin{array}{c|cccccccccc} & a_1 & a_2 & a_3 & a_4 & x_1 & x_2 & x_3 & x_4 & x_5 & x_6 \\ \hline a_1 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 1 & 0 & 0 \\ a_2 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ a_3 & 1 & 0 & 0 & 1 & 0 & 1 & 1 & 1 & 0 & 0 \\ a_4 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 \end{array}.$$

Graph and adjacency matrix representations of systems have

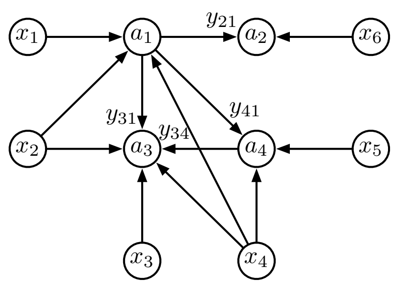

Figure 1: GRAPH REPRESENTATION OF THE EXAMPLE SYSTEM

been used extensively. Steward introduced the design structure matrix (DSM) that describes the interrelationship between design tasks [12]. Wagner introduced the functional dependence table (FDT), which identifies the variables each objective and constraint function depends on [13]. The FDT and the DSM together contain all of the information in the system adjacency matrix above.

Subproblem difficulty increases with the number of analysis functions and linking variables. Coordination problem difficulty is proportional to the number of external consistency constraints [7]. Selecting a fine partition reduces subproblem difficulty at the expense of more external consistency constraints, while coarse partitions ease coordination difficulty at the cost of more difficult subproblems. As demonstrated further below, $\mathbf{A}$ can be used to estimate how much subproblem and coordination difficulties contribute to the overall computational expense.

The following sections formalize the P/C decision problem, provide some solution techniques, and show the advantages of a simultaneous P/C decision approach. Optimal design of an electric water pump demonstrate the ideas on a physically meaningful design problem.

2 Partitioning and Coordination Decision-Making

System partitioning often follows physical or disciplinary system boundaries [13]; or product, process, or organization divisions [9] [10]. Formal partitioning approaches using FDT or DSM information are important, as they may identify non-intuitive but useful partitions [14].

Wagner defined the FDT with rows that represent constraint or objective functions, and columns that represent design, behavior, or state variables [13]. The FDT is useful for partitioning methods that seek to minimize the number of linking variables between subproblems. Behavior and state variables are coupling variables, and the design constraints represented by FDT rows include coupling variable consistency constraints. The FDT is an incidence matrix mapped to an undirected hypergraph where hyperedges represent linking variables that connect multiple functions [11]; it conceals input-output relationships and makes no distinction between shared and coupling variables. Wagner argued that defining input-output relationships a priori may limit partitioning options. In a simulation-based analysis environment, however, these relationships are often dictated a priori. In addition, failure to distinguish between shared and coupling variables eliminates the ability to exploit information flow direction for reduction of both coordination and subproblem size.

Michelena and Papalambros demonstrated the use of FDT-based spectral decomposition [11] and network reliability methods [15] for partitioning that minimizes links and exactly balances subproblem sizes. Krishnamachari and Papalambros used integer programming to generate partitions that allow some size imbalance between subsystems [16]. Chen, Ding, and Li introduced an iterative two-phase approach where the FDT is first ordered so that coupling relationships are banded along the diagonal, and then independent variable blocks and a system-wide linking variable block are formed [17]. The sequencing phase effectively reduces the 'length' of linking relationships in the system, but does not identify an optimal subproblem sequence.

Steward introduced the DSM as a tool for identifying feedback circuits in the design process and for determining a design task sequence with minimal feedback circuits [12], where rows and columns exist for all design 'parameters' and 'tasks', which are analogous to design variables and analysis functions, respectively. Rogers introduced DeMAID, a heuristic DSM-based software tool for sequencing design tasks [18], and later DeMAID/GA, which utilized a genetic algorithm to perform sequencing tasks [19]. Kroo suggested that, after sequencing has been used to minimize feedback loops, consistency constraints could be used to break any remaining feedback loops [20]. Meier, Yassine, and Browning reviewed DSM-based sequencing approaches and compared their objective functions, which primarily involved some combination of minimizing feedback, improving concurrency and modularity, or reducing computational expense [21]. These DSM-based approaches cannot use shared variables or analysis function dependance on design variables as factors in P/C decisions.

A few cases exist where some aspect of P/C interaction has been taken into account. Kusiak and Wang showed that after an FDT is used to determine appropriate subproblems, the precedence matrix (a form of the DSM) can be used to identify efficient subproblem sequences [14]. Meier et al. stated that after sequencing is performed subproblems may be more readily identified [21]. Making partitioning and coordination decisions sequentially, however, cannot account for all P/C decision interactions. It will be shown that sequential or independent approaches can fail to identify Pareto-optimal P/C options, while a simultaneous approach does not. Altus, Kroo, and Gage developed a genetic algorithm that simultaneously determined function sequence as well as 'breaks' between functions in the sequence that define the subproblems [22]. Only a single result was presented with a prescribed number of subproblems, and tradeoffs inherent to P/C decisions were not studied. Altus et al. assumed parallel subproblem solution and therefore did not define subproblem order.

The coordination decision model introduced here assumes IDF-type subproblem formulations and studies how to order subproblem solution sequences. The model is more appropriate for cases where not all subproblems can be solved in parallel. The system adjacency matrix is used to provide both FDT and DSM information while distinguishing between shared design and coupling variables.

Dealing with consistency constraints has not yet been thoroughly investigated. Different distributed design optimization formulations provide varying levels of flexibility in consistency constraint allocation: CO completely prescribes allocation; ATC allows consistency constraints for linking variables between subproblems to be assigned to any subproblem that is a common ancestor; ALD offers complete flexibility in consistency constraint allocation, an attractive feature for studying the effect of consistency constraint allocation decisions. A coordination decision model that includes consistency constraint allocation requires assumption of a particular formulation. This article examines only generic coordination formulations and does not account for consistency constraint allocation. A more sophisticated model for ALD formulations that includes both sequencing and constraint allocation is the object of future work.

2.1 Optimal Partitioning and Coordination

Partitioning and coordination decisions should minimize the complexity of the resulting distributed optimization problem. Complexity is quantified by the coordination problem size (CS) and the maximum subproblem size ($SS_{\max}$). These sizes are computed from specific partitioning and coordination information are are conflicting objectives to be minimized simultaneously. The tradeoff between CS and $SS_{\max}$ is inherent to distributed optimization problems and represents Pareto-optimum solutions. Formulas for computing CS and $SS_{\max}$ are presented next, followed by a description of optimization strategies for minimizing these quantities.

2.1.1 P/C Problem Formulation

The coordination problem size CS is defined as the total number of consistency constraints for external shared variables and feedback coupling variables, to be solved by the coordination algorithm:

$$CS = n_{\bar{x}_s m} + n_{\bar{y}f}. \tag{4}$$

The number of external shared variable consistency constraints is approximately $n_{\bar{x}_s m}$, a metric based on the number of external shared variables. The number of feedback coupling variable consistency constraints in the coordination problem is equal to the number of feedback external coupling variables $n_{\bar{y}f}$. It can be shown that the minimum number of consistency constraints required for the i-th shared variable is $n_{Pi} - 1$, where n_{Pi} is the number of subproblems that share the i-th shared variable. Therefore, the sum of $n_{Pi} - 1$ over all $n_{\bar{x}_s}$ shared variables is a reasonable approximation for the number of shared variable consistency constraints: $n_{\bar{x}_s m} = \sum_{i=1}^{n_{\bar{x}_s}} (n_{Pi} - 1)$. The reason for using $n_{\bar{y}f}$, rather than the total number of external coupling variables $n_{\bar{y}}$, is that feedback slows coordination convergence [23]. This metric penalizes existence of feedback external coupling variable consistency constraints.

The size of subproblem i, SS_i, is defined as the number of associated decision variables, consistency constraints, and analysis functions. It is assumed that IDF-type formulations are used for subproblems, meaning that no constraints are needed for internal shared variables, and one constraint is required for each internal coupling variable.

$$\begin{aligned} SS_i = \; & (n_{\bar{x}_s i} + n_{x_\ell i} + n_{yi} + n_{\bar{y}fi}) \\ & + (n_{\bar{x}_s i} + n_{yi} + n_{\bar{y}fi}) \\ & + (n_{ai}) \end{aligned} \tag{5}$$

The number of external shared variables associated with subproblem i is $n_{\bar{x}_s i}$, the number of local variables is $n_{x_\ell i}$, the number of internal coupling variables is n_{yi}, the number of coupling variables input from subproblems executed after subproblem i is $n_{\bar{y}fi}$, and the number of analysis functions is n_{ai}. $SS_{\max}$ is the maximum of all SS_i values.

2.1.2 P/C Problem Solution

Four strategies can be used to solve the P/C decision problem. In the first strategy, labeled (P,C), the P and C problems are solved independently. In the second strategy, labeled $(P \to C)$, the partitioning problem is solved first, and the resulting partition is used as a fixed parameter in coordination decision problem. The third strategy, labeled $(C \to P)$, solves the partitioning problem using a coordination method definition obtained by first solving the coordination decision problem. The fourth strategy, labeled $(P\|C)$, minimizes CS and $SS_{\max}$ simultaneously, solving the actual Pareto-optimization problem. The examples will show that the first three strategies cannot capture CS–$SS_{\max}$ tradeoff information or always identify Pareto-optimal solutions, providing evidence that interactions between partitioning and coordination decisions indeed exist and are important.

In the optimal P/C model a restricted growth string (RGS) [24], $\mathbf{p}$ of length m, is used to specify the partition by prescribing which analysis functions belong to each subproblem. The value of p_i is the subproblem that analysis function i belongs to. Redundant representations of partitions are avoided since as an RGS, $\mathbf{p}$ must satisfy:

$$p_1 = 1 \;\wedge\; p_i \le \max\{p_1, p_2, \ldots, p_{i-1}\} + 1 \tag{6}$$

Coordination decisions here are restricted to subproblem sequencing, defined by the vector $\mathbf{o}_s$, where the value of o_{si} is the evaluation position of subproblem i, and $o_{si} \neq o_{sj}\ \forall\ i, j \in \{1, 2, \ldots, N\}$. In the ($P$,$C$) and ($C \rightarrow P$) strategies coordination decisions are made without partitioning information, so it is impossible to specify a subproblem sequence and the analysis function sequence $\mathbf{o}$ is used in solving the independent coordination problems.

Two independent problems are solved in the (P,C) strategy, and the corresponding formulations are shown in Fig. 2. The independent partitioning problem seeks to find $\mathbf{p}$ that minimizes a surrogate for CS, subject to a maximum imbalance constraint (B_{allow}) and a specified number of subproblems (N_{allow}). B is the maximum subproblem size difference incurred by $\mathbf{p}$, where $SS_i - 2n_{\bar{y}fi}$ is used instead of SS_i for subproblem size since $n_{\bar{y}fi}$ depends on $\mathbf{o}_s$, which is unavailable. The value used here for B_{allow} is proportional to system size: $B_{\text{allow}} = \lfloor 0.2(m+n) \rfloor$. The surrogate used for CS that does not depend on $\mathbf{o}_s$ is $n_{\bar{x}_s m} + n_{\bar{y}}$. Forms of the independent partitioning problem have been solved previously based on FDT information [11, 14–17, 22, 25, 26]. The independent coordination decision problem seeks to find $\mathbf{o}$ that minimizes the number of feedback coupling variables n_{yf}. Since $\mathbf{p}$ is unavailable, CS and $SS_{\max}$ again cannot be used. Versions of this problem have also been solved previously [12, 18, 19, 21].

$$\begin{aligned} \min_{\mathbf{p}} \quad & n_{\bar{x}_s m} + n_{\bar{y}} \\ \text{subject to} \quad & B \leq B_{\text{allow}} \\ & N = N_{\text{allow}} \end{aligned} \qquad\qquad \min_{\mathbf{o}} \quad n_{yf}$$

Figure 2: INDEPENDENT (P,C) OPTIMIZATION APPROACH

The ($P \rightarrow C$) strategy [14] first solves the independent partitioning problem and then passes the result $\mathbf{p}^*$ as a fixed parameter to the coordination decision problem (Fig. 3). Since a partition is defined the subproblem sequence can be used as the decision vector, and both CS and $SS_{\max}$ can be used in the formulation.

$$\begin{aligned} \min_{\mathbf{p}} \quad & n_{\bar{x}_s m} + n_{\bar{y}} \\ \text{subject to} \quad & B \leq B_{\text{allow}} \\ & N = N_{\text{allow}} \end{aligned} \quad \xrightarrow{\mathbf{p}^*} \quad \begin{aligned} \min_{\mathbf{o}_s} \quad & CS \\ \text{subject to} \quad & SS_{\max} \leq SS_{\text{allow}} \end{aligned}$$

Figure 3: $P \rightarrow C$ SEQUENTIAL OPTIMIZATION

The ($C \rightarrow P$) strategy begins with solving the independent coordination decision problem for the analysis function sequence $\mathbf{o}$ (Fig. 4). Calculation of CS and $SS_{\max}$ in the second stage requires definition of a subproblem sequence. A heuristic is used here to map $\mathbf{o}$ to $\mathbf{o}_s$: subproblems are ranked in ascending order according to the lowest value of o_i in each subproblem to define the subproblem sequence.

$$\min_{\mathbf{o}} \quad n_{yf} \quad \xrightarrow{\mathbf{o}^*} \quad \begin{aligned} \min_{\mathbf{p}} \quad & CS \\ \text{subject to} \quad & SS_{\max} \leq SS_{\text{allow}} \end{aligned}$$

Figure 4: $C \rightarrow P$ SEQUENTIAL OPTIMIZATION

The ($P \| C$) strategy seeks optimal values for $\mathbf{p}$ and $\mathbf{o}_s$ simultaneously (Fig. 5). Pareto-optimal solutions are obtained by varying $SS_{\max}$ as a parameter.

$$\begin{aligned} \min_{\mathbf{p}, \mathbf{o}_s} \quad & CS \\ \text{subject to} \quad & SS_{\max} \leq SS_{\text{allow}} \end{aligned}$$

Figure 5: SIMULTANEOUS ($P \| C$) OPTIMIZATION

2.2 Examples

The four strategies were applied to two randomly generated reduced adjacency matrices to demonstrate the tradeoff between CS and $SS_{\max}$ and the interaction between partitioning and coordination decisions. The optimal P/C decision problems were all solved using exhaustive enumeration, and the appropriate constraints were varied in an effort to generate Pareto sets.

The first example has five analysis functions and seven design variables; its reduced adjacency matrix is:

$$\mathbf{A}_1 = \begin{bmatrix} 0 & 1 & 1 & 1 & 0 & 1 & 0 & 1 & 1 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 & 1 & 0 & 1 & 1 & 0 & 1 & 1 & 1 \\ 1 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 0 & 1 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 1 \end{bmatrix}.$$

Figure 6 depicts the histogram of all possible CS and $SS_{\max}$ values for an exhaustive enumeration of all possible $\mathbf{p}$ and $\mathbf{o}_s$ combinations. The CS distribution is biased toward larger values, while the $SS_{\max}$ is biased toward smaller values. This is expected since the number of possible sequences and partitions increase with N, and CS decreases with N while $SS_{\max}$ increases with N. Figure 7 plots all P/C instances for $\mathbf{A}_1$ in the CS/$SS_{\max}$ space.

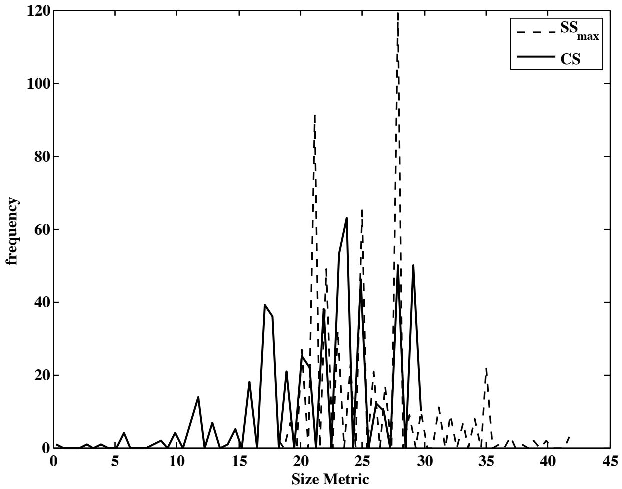

Figure 6: CS AND $SS_{\max}$ HISTOGRAMS FOR $\mathbf{A}_1$

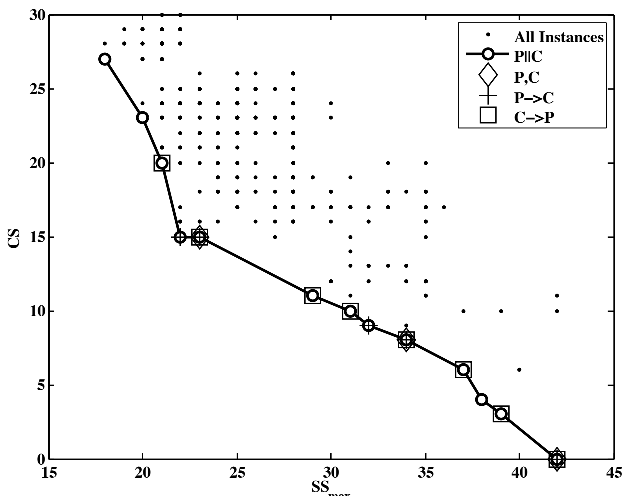

Figure 7: OPTIMIZATION RESULTS FOR $\mathbf{A}_1$

The minimum CS value of zero occurs when $N = 1$, which corresponds to a pure IDF formulation for the system design problem (with a problem size of 42). In general, distributed optimization makes sense if subproblem size can be reduced from the IDF size through partitioning without requiring a large coordination problem. This is most likely to occur when $\mathbf{A}$ is sparse. More complex products tend to have sparse adjacency matrices [10]. A minimum $SS_{\max}$ value normally occurs when each analysis function is assigned to its own subproblem but is associated with a large coordination problem.

Figure 7 also shows solutions obtained by the four different strategies. As expected, $(P\|C)$ finds all 12 Pareto points; (P,C), $(P \rightarrow C)$, and $(C \rightarrow P)$ identify 2, 4 and 7 Pareto points, respectively. These latter strategies performed well for this small example in that they identified several Pareto-optimal points. In the next slightly larger example the performance discrepancy between simultaneous and non-simultaneous approaches is more significant.

The second example has six analysis functions and ten design variables:

$$\mathbf{A}_{\mathbf{r}2} = \begin{bmatrix} 0&1&1&1&0&0&1&1&1&1&1&1&1&1&0&1 \\ 1&0&1&1&1&0&0&1&1&1&0&1&0&1&1&0 \\ 1&1&0&1&0&1&0&1&0&1&1&1&1&0&1&0 \\ 1&1&1&0&0&0&1&0&1&0&0&1&0&0&0&0 \\ 1&1&1&1&0&0&1&1&1&1&1&1&1&0&1&0 \\ 1&1&1&1&1&0&1&1&1&0&1&1&1&0&1&1 \end{bmatrix}.$$

The biases in the CS and $SS_{\max}$ distributions are now clearer in the histogram of P/C instances for $\mathbf{A}_2$ (Fig. 8). These distributions can influence the performance of algorithms other than exhaustive enumeration (such as genetic algorithms) for solving the optimal P/C problem [21].

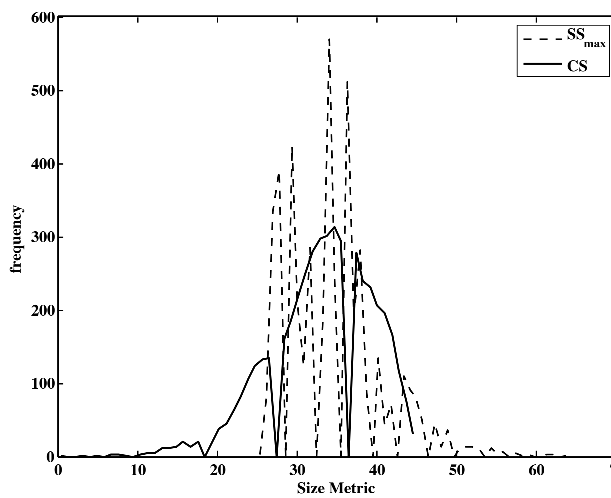

Figure 8: CS AND $SS_{\max}$ HISTOGRAMS FOR $\mathbf{A}_2$

Figure 9 shows CS and $SS_{\max}$ values for all P/C instances that exist for $\mathbf{A}_2$. $P\|C$ located all 9 Pareto points. No solutions to the non-simultaneous approaches are Pareto-optimal except for the trivial case of $N = 1$. This result is significant because if any non-simultaneous approach is used to make P/C decisions for this system, both subproblem expense and coordination problem expense could be reduced further. This sub-optimality is expected to be more pronounced as system size and complexity increases.

CS–$SS_{\max}$ tradeoff information can be used to assess whether a system is a good candidate for solution via distributed

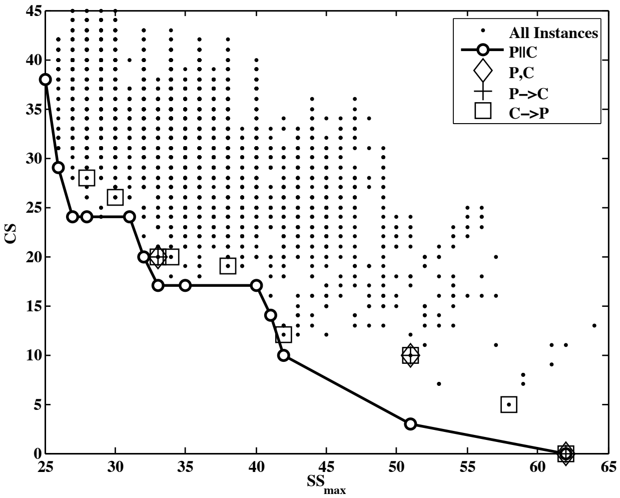

Figure 9: OPTIMIZATION RESULTS FOR $\mathbf{A}_2$

optimization since it illustrates the sensitivity of best-case solution expense to increases in partition refinement. If partitioning the system causes CS to rise sharply without appreciable $SS_{\max}$ reduction, AiO or IDF may be preferable to distributed optimization. Thus, the second example is a good candidate for distributed optimization since $SS_{\max}$ can be substantially reduced with a small associated coordination problem. In addition to helping assess whether a system should be partitioned, $P\|C$ analysis could be used to determine whether adding more sophisticated features to a coordination strategy is worthwhile if these features are accounted for in the coordination decision model.

An interesting phenomenon is displayed in Figure 9: there exists an instance where $SS_{\max} = 64$, which is greater than 62, the size of a single large subproblem. This partition cuts across a very large number of linking variables, and the subproblem order maximizes feedback. It is conceivable that some systems could exhibit this behavior for most or all P/C options, making them exceptionally poor candidates for distributed optimization.

3 Water Pump Electrification Example

A newly-developed electric water pump design problem illustrates P/C decision results for a physically meaningful system. A centrifugal water pump is used for an automotive cooling system driven by a permanent magnet DC electric motor. Traditional automotive water pumps are belt driven by the engine, and pump speed is proportional to engine speed. Such a pump must produce adequate flow and pressure even at low engine speeds. Since a pump cannot be simultaneously efficient at high and low rotational speeds, it operates at off-design flow conditions during much of its duty cycle and requires more input power than a pump driven by a constant-speed source, such as an electric motor. A motor-driven pump also has the advantage of being operated only when needed, further reducing power consumption. Electrification of belt-driven automotive components can improve fuel economy [27–29]. Surampudi et al. tested a speed-controlled electric water pump on a class-8 tractor and measured an 80% reduction in energy consumption [30].

This section describes an analysis model for an automotive electric water pump, its associated design problem, and presents P/C decision results for this application.

3.1 Analysis and Design of an Electric Water Pump

The model involves five analysis functions that compute performance metrics based on ten design variable values. Four analysis function outputs are coupling variables described in Table 1. Design variables x_1–x_5 define motor geometry, and x_6–x_{10} define pump geometry; T is computed using a thermal resistance model similar to that found in [31], adapted for permanent magnet DC motors; I and ω are computed based on fundamental DC motor equations [32, 33] adapted to the specific geometry of this motor. The pump drive torque and pressure differential are computed for a prescribed flow rate Q using fluid mechanics equations for centrifugal pumps [34]. Model details are presented in [35].

Several analysis interactions exist in this model. For example, the temperature is computed based on the motor current and speed, but the temperature affects the electrical resistance and current, and the current influences the motor speed. All model interactions are captured in the reduced adjacency matrix:

$$\mathbf{A}_3 = \begin{array}{c|cccccccccccccc} & T & I & \omega & \tau & d & d_2 & d_3 & L & \ell_c & D_2 & b & \beta_1 & \beta_2 & \beta_3 \\ \hline T & 0 & 1 & 1 & 0 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ I & 1 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ \omega & 1 & 1 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ \tau & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 \end{array}$$

Since P and τ are computed during the same analysis procedure and depend on identical quantities, P has been omitted from $\mathbf{A}_3$ for simplicity. The design problem is:

$$\begin{aligned} \min_{\mathbf{x}} \quad & P_e = VI && (7) \\ \text{subject to} \quad & P \geq P_{min} = 100\,\text{kPa} \\ & T \leq T_{\max} = 428\,\text{K} \\ & L + \ell_c \leq 0.2\,\text{m} \\ & Q = 1.55 \cdot 10^{-3}\,\text{m}^3/\text{sec} \end{aligned}$$

The source voltage V is 14.4 volts. The pressure differential and flow constraints ensure the engine is adequately cooled. The flow constraint is implicitly satisfied during the torque and pressure analysis. The temperature constraint ensures the motor wire insulation is not damaged, and the constraint on L and ℓ_c is required for packaging.

The analysis functions are very strongly coupled, and so first and second-order algorithms failed in most cases to find a consistent system analysis solution. The design problem was successfully solved using mesh adaptive direct search [36] and the IDF formulation. The minimal power consumption is 140 W, a substantial improvement over traditional water pumps of similar capacity, which consume nearly 300 W continuously [28].

Table 1: FUNCTIONS AND VARIABLES FOR THE ELECTRIC WATER PUMP PROBLEM

Analysis Functions	
$T = a_1(I, \omega, d, d_2, d_3, L, \ell_c)$	Mot. winding temp. (K)
$I = a_2(\tau, T, d, d_2, d_3, L)$	Motor current (amps)
$\omega = a_3(I, T, d, d_2, d_3, L, \ell_c)$	Motor speed (rad/sec)
$\tau = a_4(\omega, D_2, b, \beta_1, \beta_2, \beta_3)$	Pump drive torque (Nm)
$P = a_5(\omega, D_2, b, \beta_1, \beta_2, \beta_3)$	Pressure differential (kPa)

Design Variables	
$x_1 = d$	Motor wire diameter (m)
$x_2 = d_2$	Inner motor armature diameter (m)
$x_3 = d_3$	Outer motor armature diameter (m)
$x_4 = L$	Motor armature length (m)
$x_5 = \ell_c$	Motor commutator length (m)
$x_6 = D_2$	Pump impeller diameter (m)
$x_7 = b$	Pump impeller blade width (m)
$x_8 = \beta_1$	Pump blade angle at inlet (rad)
$x_9 = \beta_2$	Pump blade angle at outlet (rad)
$x_{10} = \beta_3$	Pump diffuser inlet angle (rad)

3.2 Optimal P/C Solution Results

The optimal P/C problem was solved using all four strategies. As can be seen in Fig. 10, $(P\|C)$ and $(P \rightarrow C)$ identified all four Pareto points, while (P,C) and $(C \rightarrow P)$ were only able to identify one Pareto point (the trivial solution with $N = 1$).

Of particular interest is the initial low sensitivity of CS to increased N. $SS_{\max}$ can be reduced from 28 to 19 with a CS of 1, making this system an excellent candidate for distributed optimization. Only the $(P\|C)$ and $(P \rightarrow C)$ strategies can reveal this low sensitivity.

The matrix $\mathbf{A}_3$ represents a physical system, so P/C decisions made based on engineering intuition can be compared to optimal P/C modeling results. Dividing the system into motor and pump-related functions corresponds to the partition $\mathbf{p} = [1,1,1,2]$. If the motor subproblem is solved first, then $CS = 1$ and $SS_{\max} = 20$, a good but suboptimal solution. Using a model-derived partition $\mathbf{p} = [1,2,3,4]$ as a starting point to solve coordination problem defined in Fig. 3 for the optimal sequence, the solution $\mathbf{o}_s^* = [4,3,2,1]$ yields $CS = 12$ and $SS_{\max} = 13$, which

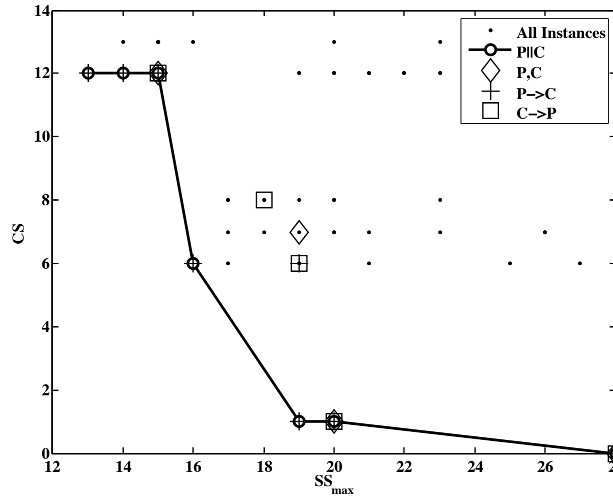

Figure 10: OPTIMAL P/C RESULTS FOR PUMP PROBLEM

is a Pareto point. In this simple example, intuitive and semi-intuitive approaches are rather effective, but cannot quantify the tradeoff between CS and $SS_{\max}$. Much larger systems are likely to realize greater benefits from the $(P\|C)$ strategy, but algorithms more sophisticated than exhaustive enumeration will be required in such implementations, a topic for future work.

4 Conclusion

We introduced a formal approach for simultaneous partitioning and coordination decision-making, in order to investigate the suitability of a system for distributed optimization. The approach quantifies P–C tradeoffs by computing Pareto optima for minimum subproblem size and coordination problem size. The problem-size metrics proposed here captured P/C interactions in the examples successfully. Other metrics can be used instead if desirable. Simultaneous P/C optimization can lead to superior decomposition solutions. Comparison to non-simultaneous strategies confirmed the existence of P/C decision interaction, and demonstrated the value of a simultaneous approach. Exhaustive enumeration was used to generate results for small examples, and a simplified coordination decision model incorporated only subproblem sequencing. An improved coordination decision model that accounts for consistency constraint allocation is a next step in this research. Exhaustive enumeration must be replaced by more efficient algorithms that can generate the Pareto-optimal solutions for larger systems, a topic currently under investigation.

Acknowledgments

This work was partially supported by a US NSF Graduate Research Fellowship and by the Automotive Research Center, a US Army Center of Excellence at the University of Michigan. This support is gratefully acknowledged.

REFERENCES

[1] Cramer, E. J., Dennis, J. E., Frank, P. D., Lewis, R. M., and Shubin, G. R., 1994. "Problem formulation for multidisciplinary optimization". *SIAM Journal of Optimization,* **4**, pp. 754–776.

[2] Allison, J., Kokkolaras, M., and Papalambros, P., 2007. "On selection of single-level formulations for complex system design optimization". *To appear in Journal of Mechanical Design, Transactions of the ASME,* **129**(9).

[3] Balling, R. J., and Sobieszczanski-Sobieski, J., 1996. "Optimization of coupled systems: A critical overview of approaches". *AIAA Journal,* **34**(1), pp. 6–17.

[4] Braun, R., 1996. "Collaborative optimization: An architecture for large-scale distributed design". PhD thesis, Stanford University, April.

[5] Kim, H. M., Michelena, N. F., Papalambros, P. Y., and Jiang, T., 2003. "Target cascading in optimal system design". *Journal of Mechanical Design, Transactions of the ASME,* **125**(3), pp. 474–480.

[6] Tosserams, S., 2004. "Analytical target cascading: convergence improvement by sub-problem post-optimality sensitivities". Master's thesis, Eindhoven University of Technology.

[7] Tosserams, S., Etman, L., and Rooda, J., 2007. "An augmented Lagrangian decomposition method for quasi-separable problems in MDO". *To appear in Structural and Multidisciplinary Optimization*.

[8] Sobieszczanski-Sobieski, J., and Haftka, R., 1997. "Multidisciplinary aerospace design optimization: Survey of recent developments". *Structural Optimization,* **14**(1), pp. 1–23.

[9] Chen, B., Liu, J., Zhong, Y., and Zhou, J., 2000. "Properties of the coupled factors in MDO and their application in collaborative optimization". In the Proceedings of DETC'00, ASME 2000 Design Engineering Technical Conferences and Computers and Information in Engineering Conference, DETC2000/DAC14516.

[10] Kusiak, A., and Larson, N., 1995. "Decomposition and representation methods in mechanical design". *Journal of Mechanical Design, Transactions of the ASME,* **117B**, pp. 17–24.

[11] Michelena, N., and Papalambros, P., 1997. "A hypergraph framework for optimal model-based decomposition of design problems". *Computational Optimization and Applications,* **8**(2), pp. 173–196.

[12] Steward, D., 1981. "The design structure system: a method for managing the design of complex systems". *IEEE Transactions on Engineering Management,* **EM-28**(3), pp. 71–4.

[13] Wagner, T., 1993. "A general decomposition methodology for optimal system design". PhD thesis, University of Michigan.

[14] Kusiak, A., and Wang, J., 1993. "Decomposition of the design process". *Journal of Mechanical Design, Transactions of the ASME,* **115**(4), pp. 687–695.

[15] Michelena, N., and Papalambros, P., 1995. "Network reliability approach to optimal decomposition of design problems". *Journal of Mechanical Design, Transactions of the ASME,* **117**(3), pp. 433–440.

[16] Krishnamachari, R. S., and Papalambros, P. Y., 1997. "Optimal hierarchical decomposition synthesis using integer programming". *Journal of Mechanical Design, Transactions of the ASME,* **119**(4), pp. 440–447.

[17] Chen, L., Ding, Z., and Li, S., 2005. "A formal two-phase method for decomposition of complex design problems". *Journal of Mechanical Design, Transactions of the ASME,* **127**(2), pp. 184–195.

[18] Rogers, J., 1989. DeMAID—a design manager's aid for intelligent decomposition user's guide. Tech. Rep. TM-101575, NASA.

[19] Rogers, J., 1996. "DeMAID/GA - an enhanced design manager's aid for intelligent decomposition (genetic algorithms)". In the Proceedings of the 6th AIAA/USAF/NASA/ISSMO Symposium on Multidisciplinary Analysis and Optimization, AIAA-96-4157-CP, pp. 1497–1504.

[20] Kroo, I., Altus, S., Braun, R., Gage, P., and Sobieski, I., 1994. "Multidisciplinary optimization methods for aircraft preliminary design". In the Proceedings of the 5th AIAA/USAF/NASA/ISSMO Symposium on Multidisciplinary Analysis and Optimization, AIAA-94-4325-CP, pp. 697–707.

[21] Meier, C., Yassine, A., and Browning, T., 2007. "Design process sequencing with competent genetic algorithms". *Journal of Mechanical Design, Transactions of the ASME,* **129**(6), pp. 566–585.

[22] Altus, S., Kroo, I., and Gage, P., 1996. "Genetic algorithm for scheduling and decomposition of multidisciplinary design problems". *Journal of Mechanical Design, Transactions of the ASME,* **118**(4), pp. 486–489.

[23] Bertsekas, D. P., and Tsitsiklis, J. N., 1997. *Parallel and Distributed Computation: Numerical Methods*. Athena Scientific, Belmont, MA.

[24] Stanton, D., and White, D., 1986. *Constructive Combinatorics*. Springer-Verlag, New York.

[25] Michelena, N., and Papalambros, P., 1995. "Optimal model-based decomposition of powertrain system design". *Journal of Mechanical Design, Transactions of the ASME,*

117(4), pp. 499–505.

[26] Chen, L., and Li, S., 2005. "Analysis of decomposability and complexity for design problems in the context of decomposition". *Journal of Mechanical Design, Transactions of the ASME,* **127**(4), pp. 545–557.

[27] McCleer, P., 1995. "Electric drives for pump, fan, and compressor loads in automotive applications". In the Proceedings of the IEEE International Symposium on Industrial Electronics, pp. 80–85.

[28] Kassakian, J. G., 1996. "Future of power electronics in advanced automotive electrical systems". In IEEE Annual Power Electronics Specialists Conference, pp. 7–14.

[29] Kokkolaras, M., Mourelatos, Z., Louca, L., Filipi, Z., Delagrammatikas, G., Stefanopoulou, A., Papalambros, P. Y., and Assanis, D., 2006. "Design under uncertainty and assessment of performance reliability of a dual-use medium truck with hydraulic-hybrid powertrain and fuel cell auxiliary power unit". *SAE 2005 Transactions: Journal of Passenger Cars - Mechanical Systems*. (Selected from the Proceedings of the SAE World Congress, April 11-14, 2005, Detroit, Michigan, paper no. 2005-01-1396).

[30] Surampudi, B., Redfield, J., Ray, G., Montemayor, A., Walls, M., McKee, H., Edwards, T., and Lasecki, M., 2005. "Electrification and integration of accessories on a class-8 tractor". In the Proceedings of the SAE World Congress, SAE-2005-01-0016.

[31] Mellor, P., Roberts, D., and Turner, D., 1991. "Lumped parameter thermal model for electrical machines of TEFC design". *IEE Proceedings B: Electric Power Applications,* **138**(5), pp. 205–218.

[32] Kenjo, T., and Nagamori, S., 1985. *Permanent-Magnet and Brushless DC Motors*. Oxford University Press, New York.

[33] Gieras, J. F., and Wing, M., 2002. *Permanent Magnet Motor Technology*. Marcel Dekker, Inc., New York.

[34] Tuzson, J., 2000. *Centrifugal Pump Design*. John Wiley and Sons, New York.

[35] Allison, J., 2007. Design optimization model for an automotive electric water pump. Tech. rep., University of Michigan, Optimal Design Laboratory. http://ode.engin.umich.edu/repository/TR2007.Allison.pdf.

[36] Abramson, M., Audet, C., and Dennis Jr., J., 2006. "Nonlinear programing with mesh adaptive direct searches". *SIAG/Optimization Views-and-News,* **17**(1), pp. 2–11.