Optimal Partitioning and Coordination Decisions in Decomposition-Based Design Optimization

James T. Allison
Senior Applications Engineer
The MathWorks, Inc.,
Natick, MA 01760
e-mail: james.allison@mathworks.com

Michael Kokkolaras
Associate Research Scientist
e-mail: mk@umich.edu

Panos Y. Papalambros
Professor
e-mail: pyp@umich.edu

Department of Mechanical Engineering,
University of Michigan,
G.G. Brown Building,
Ann Arbor, MI 48109

The solution of complex system design problems using 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 and coordination strategies. Promising preliminary results with small test problems are presented. The approach is illustrated on an electric water pump design problem.
[DOI: 10.1115/1.3178729]

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 from a set of viable alternatives. 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 a partitioning and coordination decision-making model is formulated as an optimization problem and solved for test problems. Initial results indicate that accounting for the interaction between partitioning and coordination can lead to better decomposition-based optimization strategies.

1.1 Decomposition-Based System Design. The system design problems considered here involve multidisciplinary coupled analyses (e.g., a set of coupled computer-aided engineering simulations) where input/output properties are assumed to be known precisely. The vector of quantities computed by the jth analysis function and required as input to the ith analysis function is termed as the analysis coupling variable $\mathbf{y}_{ij}$. The vector of all coupling variable 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 ith analysis function as $\mathbf{a}_i(\mathbf{x}_i,\mathbf{y}_i)$. Analysis functions can be objective, constraint, or intermediate functions in the system design optimization problem. Design variables that are inputs to $\mathbf{a}_i(\mathbf{x}_i,\mathbf{y}_i)$ only are 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 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 $\mathbf{a}_i(\mathbf{x}_i,\mathbf{y}_i)$ 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

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

is satisfied, where the Boolean matrix $\mathbf{S}_{ij}$ selects the components of $\mathbf{a}_j$ that correspond to $\mathbf{y}_{ij}$. 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 selection matrix that extracts the components of $\mathbf{a}(\mathbf{x},\mathbf{y})$ that correspond to $\mathbf{y}$. 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

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

where $\mathbf{y}_{\mathrm{p}}(\mathbf{x})$ 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

Contributed by the Design Automation Committee of ASME for publication in the JOURNAL OF MECHANICAL DESIGN. Manuscript received May 17, 2007; final manuscript received April 15, 2009; published online July 27, 2009. Review conducted by Timothy W. Simpson. Paper presented at the ASME 2007 Design Engineering Technical Conferences and Computers and Information in Engineering Conference (DETC2007), Las Vegas, NV, September 4–7, 2007.

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 Sobieszczanski-Sobieski [3] suggested a hybrid approach that uses function sequencing to satisfy feedforward coupling relationships and equality constraints to satisfy feedback coupling relationships.

Decomposition-based design 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. System partitioning results in links between subproblems; the number and nature of these links can be a dominant factor in the solution process. In many cases the subproblems may be solved in parallel; henceforth we assume serial computation. Additional work investigates partitioning and coordination decisions for a specific parallel system optimization formulation [4]. Equality constraints or penalty functions may be employed 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 $\bar{\mathbf{y}}_{ij}$. External shared and coupling variables for subproblem i comprise its set of external linking variables $\bar{\mathbf{z}}_i$. Independent subproblem solution requires local copies of both external coupling and 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., quasiseparable decomposition approach of Haftka and Watson [5], concurrent subsystem optimization (CSSO) [6], and collaborative optimization (CO) [7]), fixed point iteration (e.g., analytical target cascading (ATC) [8]), Newton's method (e.g., ATC [9]), and penalty methods (e.g., CSSO [10], ATC, and augmented Lagrangian coordination (ALC) [11]). Note that Rodriguez et al. [10] introduced the use of augmented Lagrangian penalty relaxation with the method of multipliers as a coordination method for decomposition-based design optimization; this first approach was based on CSSO with response surface approximations for subproblems. Tosserams et al. [11] later introduced an augmented Lagrangian formulation for ATC [12] and then ALC. Decomposition-based methods are appropriate when systems are large and sparsely connected [13], when the design environment is distributed [14], or when specialized optimization algorithms can be exploited for solving particular subproblems [15,16]. The techniques introduced in this article can be used to assess quantitatively whether decomposition-based optimization is appropriate for a particular problem.

1.2 Partitioning and Coordination. A method for decomposition-based design optimization (i.e., a *decomposition method*) is defined here to include both a system partition and a coordination strategy [17]. To apply a decomposition method we must first determine a system partition and a coordination strategy. The partitioning problem (P) is to decide which of m analysis functions should be clustered into each of the N subproblems. The coordination decision problem (C) is to specify a method for satisfying system consistency requirements; this typically involves consistency constraint management and a strategy for guiding repeated subproblem solutions toward system optimality and consistency. Decomposition methods in this article are assumed to be penalty relaxation methods where consistency constraints are only satisfied at convergence and design constraints are satisfied at every subproblem solution. This class includes the ATC and ALC formulations. The system design problems are assumed to be quasiseparable, i.e., subproblems may share design variables but not design constraints. Proofs exist for ATC [18] and ALC [11,19] that demonstrate convergence under standard assumptions, such as convexity, and show that the solution to the decomposed problem is identical to that of the original design problem. Convergence and equivalence to AiO for a few other methods, such as those proposed by Rodriguez et al. [10] and Haftka and Watson [5], have been proven under looser conditions, but these methods relax design constraints and therefore do not fall under the class described above. It is important to acknowledge that several alternative decomposition methods, such as CO, have been shown to have theoretical and practical problems [20], highlighting the importance of the methods described above with proven convergence and equivalence properties. The coordination algorithm is assumed to be a fixed point iteration (FPI), and therefore decomposition method convergence is subject to FPI convergence conditions [2] (the a priori verification of which is very difficult in most practical cases). Subproblem solution sequence can influence convergence rate significantly [21] and is a defining property of the coordination strategy. This article studies partitioning decisions as well as the subproblem sequence aspect of coordination. Investigation of consistency constraint management is addressed in Ref. [4], and alternative coordination algorithms are a topic for future work.

Analysis function interactions and dependence on design variables are important factors in partitioning and coordination decisions. Matrix representations can effectively describe relationships in system design problems. Steward [22] proposed the structural matrix (SM) for illustrating relationships in systems of equations. SM rows correspond to equations and columns to variables that appear in the equations. The SM is useful for making partitioning decisions but lacks directionality information required for determining solution sequence. An output set is required to define directionality. Steward [23] later introduced the design structure matrix (DSM) that describes the inter-relationship between design elements, rather than equations and variables. These design elements were originally described as either design tasks or parameters, although later DSM approaches typically limit design elements to either design tasks or parameters but not both. The DSM is a square adjacency matrix where the elements represented by rows and columns are identical. The DSM is well suited for describing information flow direction and has been used extensively in ordering design tasks to reduce feedback loops [24]. A DSM could be used to make combined partitioning and sequence decisions if its design elements included both analysis functions and design variables. Another related matrix representation is the functional dependence table (FDT) introduced by Wagner [17]. The FDT is similar to the SM but is intended specifically for partitioning large equation-based design optimization problems. Instead of mapping variables to equations, the FDT maps design and coupling variables to objective and constraint functions. It lacks directionality information like the SM and cannot be used for sequencing unless an output set is defined.

Other matrices associated with engineering design, but normally not used in P/C decisions, include the relation matrix (RM) and correlation matrix (CM) from quality function deployment [25], and the design matrix (DM) from axiomatic design [26]. The RM maps product engineering characteristics to customer requirements, and the CM describes correlations between engineering characteristics. The DM maps design variables to functional requirements and is intended for evaluating independence of functional requirements when comparing design concepts.

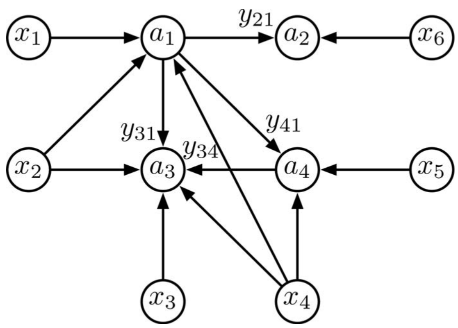

Fig. 1 Digraph representation of the example system

An example system is used to illustrate the matrix representation used in this article. Consider the following system of analysis functions:

$$a_1(x_1,x_2,x_4)$$

$$a_2(x_6,y_{21})$$

$$a_3(x_2,x_3,x_4,y_{31},y_{34})$$

$$a_4(x_4,x_5,y_{41})$$

The analysis functions are interdependent, and x_2 and x_4 are shared design variables. A directed graph (digraph) represents system relationships effectively (Fig. 1). If a system's digraph contains a cycle, then feedback coupling exists. Only simulation-based design problems are considered here. These have well-defined input-output relationships; therefore, a digraph is an appropriate representation.

This system could be represented with an SM and output set, or a DSM with both analysis functions and design variables. Such a DSM, shown below with the design elements in arbitrary order, is the transpose of the system digraph's adjacency matrix:

DSM =

	x_1	x_2	a_1	a_3	x_3	a_2	a_4	x_4	x_5	x_6
x_1	0	0	0	0	0	0	0	0	0	0
x_2	0	0	0	0	0	0	0	0	0	0
a_1	1	1	0	0	0	0	0	1	0	0
a_3	0	1	1	0	1	0	1	1	0	0
x_3	0	0	0	0	0	0	0	0	0	0
a_2	0	0	1	0	0	0	0	0	0	1
a_4	0	0	1	0	0	0	0	1	1	0
x_4	0	0	0	0	0	0	0	0	0	0
x_5	0	0	0	0	0	0	0	0	0	0
x_6	0	0	0	0	0	0	0	0	0	0

Design variables are independent quantities; therefore, rows representing them are zero and can be omitted without loss of information. Organizing the matrix such that analysis functions appear before design variables aid visualization of system structure. In addition, ordering functions and variables by index value is convenient for calculating metrics used in P/C decisions. This condensed and reordered matrix is termed the reduced adjacency matrix $\mathbf{A}$. The system adjacency matrix is $[\mathbf{A}^{\mathrm{T}},\mathbf{0}]$. The reduced adjacency matrix for the example system is

A =

	a_1	a_2	a_3	a_4	x_1	x_2	x_3	x_4	x_5	x_6
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

Subproblem solution difficulty typically increases with the number of analysis functions, design variables, and linking variables, although analysis function properties also influence difficulty. Similarly, coordination problem solution expense normally increases with the number of external consistency constraints [19]. Fine partitions reduce 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 in Secs. 2 and 3, **A** can be used to estimate how much subproblem and coordination difficulties contribute to the overall computational expense. As with most matrix representations, **A** describes the existence, but not the nature, of functional relationships. The nature of these interactions influences convergence rate. More sophisticated models involving function sensitivities over the analysis and design space are too computationally expensive to justify calculation. Initial findings indicate that a P/C decision method based on **A** is reasonably efficient, and the resulting P/C decisions provide an advantage when solving the system optimization problem.

1.3 Partitioning and Coordination Decision-Making. Subjective techniques for P/C decisions are in common use. System partitioning often follows physical or disciplinary system boundaries [17], or product, process, or organization divisions [14,15]. Qualitative guidance for coordination method selection exists in the literature [2,3,13,27]. Formal P/C decision techniques are important, as they may identify nonintuitive and potentially advantageous system partitions and coordination strategies [28]. A review of established P/C decision techniques follows; most center on either partitioning or sequence decisions, a few consider some aspect of P/C interaction.

Michelena and Papalambros [16] demonstrated the use of spectral and network reliability methods [29] to obtain partitions that minimize external linking variables and exactly balance subproblem sizes. Krishnamachari and Papalambros [30] used integer programming to generate partitions that allow some subproblem size imbalance. Chen et al. [31] 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 systemwide linking variable block is formed.

Steward used the DSM with a "tearing" algorithm to order design elements so that blocks of closely coupled tasks can be identified, forming a type of partition. In this approach partitions depend on sequence decisions; partitioning decisions cannot be made independently, and superior P/C decisions may be overlooked. Rogers [32] introduced DEMAID, a heuristic DSM-based software tool for sequencing design tasks, and later DEMAID/GA [33], which utilized a genetic algorithm to perform sequencing tasks. Kroo et al. [34] suggested that, after sequencing was used to minimize feedback loops, consistency constraints could be used to break any remaining feedback loops. Meier et al. [35] 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.

A few approaches have accounted for some aspect of P/C interaction. Kusiak and Wang [28] demonstrated a method that first partitions a system based on its FDT, and then identifies an efficient subproblem sequence using a precedence matrix. This is similar to Steward's SM approach, except that Steward first determined a sequence and then identified a partition. Meier et al. [35] also described how partitions can be identified after a sequence is defined. A sequential P/C decision process, 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 et al. [36] developed a genetic algorithm that simultaneously determined function sequence as well as "breaks" between functions that form a partition. Only a single result was presented with a prescribed number of subproblems, P/C decision tradeoffs were not studied, and subproblem order was not defined since parallel subproblem solution was assumed.

Recall that coordination decisions involve both subproblem sequence and consistency constraint management. Different decomposition methods provide varying levels of flexibility in how consistency constraints may be allocated in a decomposition method: CO prescribes allocation completely; ATC allows consistency constraints for linking variables between subproblems to be assigned to any subproblem that is a common ancestor; and ALC 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 ALC formulations that includes both sequencing and constraint allocation is presented in Ref. [4].

The P/C decision model introduced here assumes IDF-type subproblem formulations and a fixed-point iteration coordination algorithm; the model aids study of interactions between partitioning and subproblem solution sequence decisions. P/C decisions are based on metrics computed using the system reduced adjacency matrix. P/C decision results are optimal with respect to the P/C decision model.

Sections 2 and 3 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 demonstrates the ideas on a physically meaningful design problem.

2 Partitioning and Coordination Decision Model

Partitioning and coordination decisions should minimize the complexity of the resulting system optimization problem. Complexity is approximated here by the coordination problem size (CS) and the maximum subproblem size ($SS_{\max}$). These sizes are computed from specific partitioning and coordination information and are conflicting objectives to be minimized simultaneously. The tradeoff between CS and $SS_{\max}$ is inherent to decomposition methods and represents P/C-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 *P/C* Problem Formulation. The coordination problem and subproblem size metrics were derived based on a distributed optimization formulation, such as ATC or ALC, where consistency is managed using a penalty relaxation method and the subproblems are coordinated using fixed-point iteration. 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 following 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 ith external shared variable is $n_{Pi}-1$, where n_{Pi} is the number of subproblems that share the ith external shared variable. Therefore, the sum of $n_{Pi}-1$ over all $n_{\bar{x}_s}$ external shared variables is a reasonable approximation for the number of external shared variable consistency constraints: $n_{\bar{x}_s m}=\sum_{i=1}^{n_{\bar{x}_s}}(n_{Pi}-1)$. The reason $n_{\bar{y}f}$ is used instead of the total number of external coupling variables $n_{\bar{y}}$ is to penalize feedback, which slows coordination convergence [37]. CS does not model how coordination problem size depends on consistency constraints allocation, and is therefore an approximation.

$$\begin{array}{ll} \min\limits_{\mathbf{p}} & n_{\bar{x}_s m} + n_{\bar{y}} \\ \text{subject to} & B \le B_{\text{allow}} \\ & N = N_{\text{allow}} \end{array} \qquad\qquad \min\limits_{\mathbf{o}} \quad n_{yf}$$

Fig. 2 Independent (*P*, *C*) optimization approach

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

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

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.

2.2 *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 $(P \rightarrow C)$, the partitioning problem is solved first, and the resulting partition is used in solving the coordination decision problem. The third strategy, labeled $(C \rightarrow P)$, 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 the $CS-SS_{\max}$ 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) [38] $\mathbf{p}$ of length m is used to specify the partition by prescribing which analysis function belongs 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 the following:

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

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 $(C \rightarrow P)$ strategy 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 instead.

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

$$\boxed{\begin{array}{ll}\min\limits_{\mathbf{p}} & n_{\bar{x}_s m} + n_{\bar{y}} \\ \text{subject to} & B \le B_{\text{allow}} \\ & N = N_{\text{allow}}\end{array}} \xrightarrow{\mathbf{p}^*} \boxed{\begin{array}{ll}\min\limits_{\mathbf{o}_s} & CS \\ \text{subject to} & SS_{\max} \le SS_{\text{allow}}\end{array}}$$

Fig. 3 ***P*→*C* sequential optimization**

$-2n_{\bar{y}fi}$ 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 were solved previously [16,28–31,36,39,40]. 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 the independent coordination problem were also solved previously [23,32,33,35].

The $(P \to C)$ strategy [28] 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.

The $(C \to P)$ 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.

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_{allow} as a parameter.

2.3 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. In other words, every point represents a different system partition and coordination strategy option. In many cases several $\mathbf{p}$ and $\mathbf{o}_s$ combinations result in the same $CS/SS_{\max}$ values.

$$\boxed{\begin{array}{ll}\min\limits_{\mathbf{o}} & n_{yf}\end{array}} \xrightarrow{\mathbf{o}^*} \boxed{\begin{array}{ll}\min\limits_{\mathbf{p}} & CS \\ \text{subject to} & SS_{\max} \le SS_{\text{allow}}\end{array}}$$

Fig. 4 ***C*→*P* sequential optimization**

$$\boxed{\begin{array}{ll}\min\limits_{\mathbf{p},\mathbf{o}_s} & CS \\ \text{subject to} & SS_{\max} \le SS_{\text{allow}}\end{array}}$$

Fig. 5 **Simultaneous (*P*‖*C*) optimization**

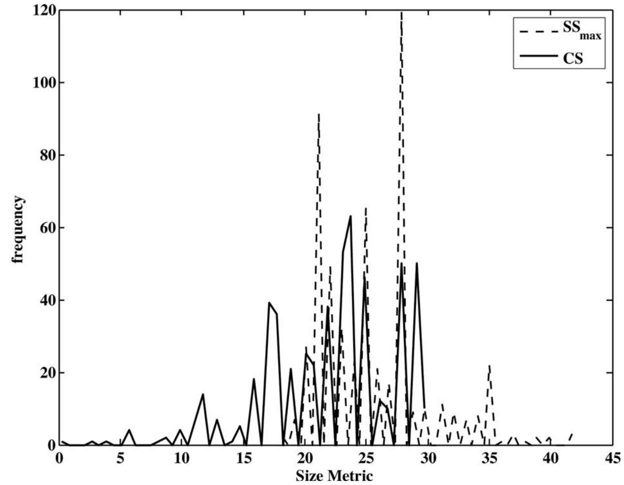

Fig. 6 ***CS*** **and** ***SS*****max** **histograms for A1**

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, decomposition methods make 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. Complex products tend to have sparse adjacency matrices [15]. 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), $(P \to C)$, and $(C \to P)$ 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. A parametric study on B_{allow} values revealed that increasing allowed imbalance for the (P, C) and $(P \to C)$ approaches initially improves the number of Pareto points identified, but increasing B_{allow}

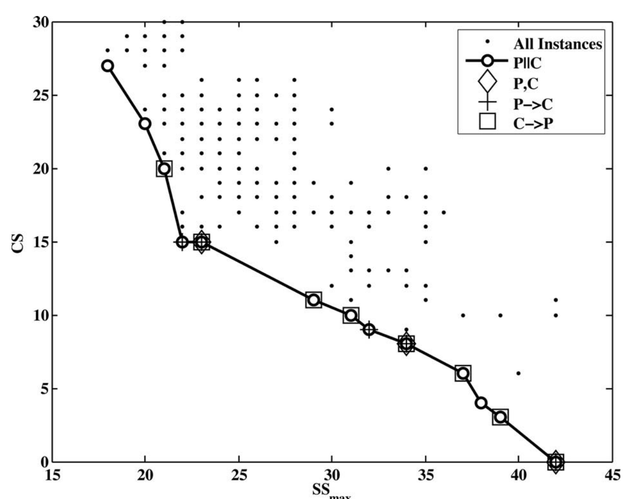

Fig. 7 **Optimization results for A1**

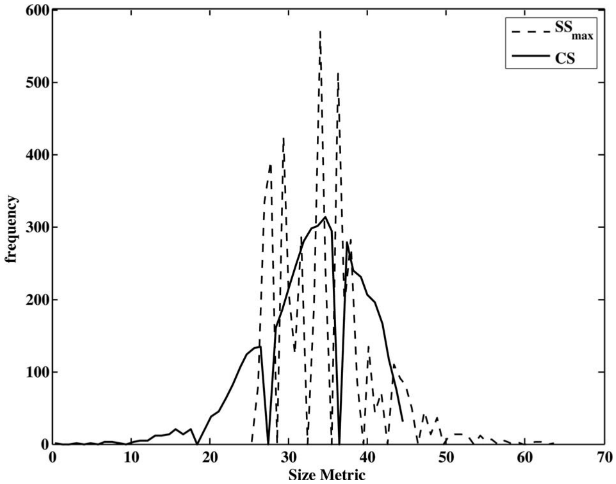

Fig. 8 CS and SS_{max} histograms for A_2

Table 1 Functions and variables for the electric water pump problem

Analysis functions	
$T=a_1(I,\omega,d,d_2,d_3,L,\ell_c)$	Motor winding temperature (K)
$I=a_2(\tau,T,d,d_2,d_3,L)$	Motor current (A)
$\omega=a_3(I,T,d,d_2,d_3,L,\ell_c)$	Motor speed (rad/s)
$\tau=a_4(\omega,D_2,b,\beta_1,\beta_2,\beta_3)$	Pump drive torque (N m)
$P=a_5(\omega,D_2,b,\beta_1,\beta_2,\beta_3)$	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)

much beyond $\lfloor 0.2(m+n)\rfloor$ does not continue to improve results. In all cases nonsimultaneous approaches identified only a fraction of the Pareto set. In the next slightly larger example the performance discrepancy between simultaneous and nonsimultaneous approaches is more significant. The second example has six analysis functions and ten design variables.

$$\mathbf{A}_2=\begin{bmatrix} 0&1&1&1&0&0&1&1&1&1&1&1&1&1&0&1\\ 1&0&1&1&1&0&0&1&1&1&0&1&0&1&1&0\\ 1&1&0&1&0&1&0&1&0&1&1&1&1&0&1&0\\ 1&1&1&0&0&0&1&0&1&0&0&1&0&0&0&0\\ 1&1&1&1&0&0&1&1&1&1&1&1&1&0&1&0\\ 1&1&1&1&1&0&1&1&1&0&1&1&1&0&1&1 \end{bmatrix}$$

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 [41]) for solving the optimal P/C problem [35].

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

$CS-SS_{max}$ tradeoff information can be used to assess system suitability for solution via a decomposition method since it illustrates the sensitivity of best-case solution expense to increases in partition refinement. If increasing N causes CS to rise sharply without appreciable SS_{max} reduction, AiO or IDF may be preferable. Thus, the second example is a good candidate for decomposition-based design optimization.

An interesting phenomenon is evident in Fig. 9: There exists an instance where $SS_{max}=64$, which is greater than 62, the size of a single large subproblem. The corresponding 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 a decomposition-based approach.

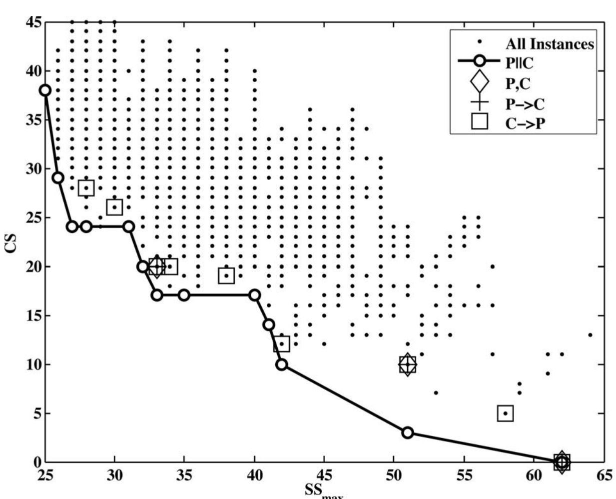

Fig. 9 Optimization results for A_2

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 [42–44]. Surampudi et al. [45] tested a speed-controlled electric water pump on a class-8 tractor and measured an 80% reduction in energy consumption.

Section 3.1 describes an analysis model for an automotive electric water pump, its associated design problem, and Sec. 3.2 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 Ref. [46] adapted for permanent magnet dc motors; I and ω are computed based on fundamental dc motor equations [47,48] 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 [49]. Model details are presented in Ref. [50]. The pump design problem is not globally convex, and it is not known whether it satisfies convergence and equivalence requirements for ALC. This article focuses on coupled partitioning and coordination decisions, not equivalence of decomposed problems with AiO. Convergence and equivalence analysis is left for future work.

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 following 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 objective is to minimize electrical power consumption (P_e), and the optimization formulation is as follows:

$$\min_{\mathbf{x}} P_e = VI$$

$$\text{subject to} \quad P \geq P_{\min} = 100 \ \text{kPa}$$

$$T \leq T_{\max} = 428 \ \text{K} \tag{7}$$

$$L + \ell_c \leq 0.2 \ \text{m}$$

$$Q = 1.55 \cdot 10^{-3} \ \text{m}^3/\text{s}$$

The source voltage V is 14.4 V. The pressure differential and flow constraints ensure the engine is adequately cooled. The flow constraint is satisfied implicitly 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 the 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 [51] and the IDF formulation. One possible alternative solution algorithm is DIRECT [52,53]. The minimal power consumption is 140 W, a substantial improvement over traditional water pumps of similar capacity, which consume nearly 300 W continuously [43].

3.2 *P/C* Decision Results. The P/C decision problem was solved using all four optimization 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 decomposition-based design 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 with opti-

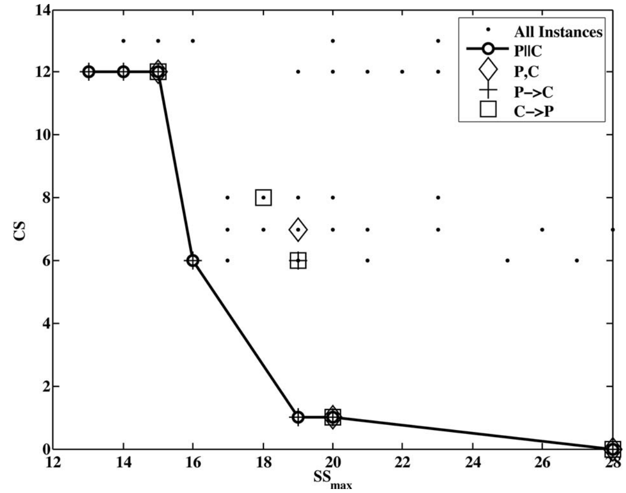

Fig. 10 Optimal *P/C* results for pump problem

mal 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 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 [41].

4 Conclusion

We introduced a formal approach for simultaneous partitioning and coordination decision-making to investigate the suitability of a system for decomposition-based design 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 desired. Simultaneous P/C decision-making can lead to superior decomposition solutions. Comparison to nonsimultaneous 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 the 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.

Acknowledgment

This work was partially supported by the U.S. NSF Graduate Research Fellowship and by the Automotive Research Center, a U.S. Army Center of Excellence at the University of Michigan. This support is gratefully acknowledged. The work of the first author was conducted while at the University of Michigan.

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 J. Optim., **4**, pp. 754–776.

[2] Allison, J. T., Kokkolaras, M., and Papalambros, P. Y., 2007, "On Selection of

Single-Level Formulations for Complex System Design Optimization," ASME J. Mech. Des., **129**(9), pp. 898–906.
[3] Balling, R. J., and Sobieszczanski-Sobieski, J., 1996, "Optimization of Coupled Systems: A Critical Overview of Approaches," AIAA J., **34**(1), pp. 6–17.
[4] Allison, J. T., and Papalambros, P. Y., 2008, "Consistency Constraint Allocation in Augmented Lagrangian Coordination," Proceedings of the DETC'08, ASME 2008 Design Engineering Technical Conferences and Computers and Information in Engineering Conference, Brooklyn, NY, Aug. 3–6, Paper No. DETC2008-49283.
[5] Haftka, R., and Watson, L., 2005, "Multidisciplinary Design Optimization With Quasiseparable Subsystems," Optim. Eng., **6**, pp. 9–20.
[6] Sobieszczanski-Sobieski, J., 1988, "Optimization by Decomposition: A Step From Hierarchic to Non-Hierarchic Systems," Proceedings of the Second NASA/USAF Symposium on Recent Advances in Multidisciplinary Analysis and Optimization, Hampton, VA, Sept. 28–30.
[7] Braun, R. D., 1996, "Collaborative Optimization: An Architecture for Large-Scale Distributed Design," Ph.D. thesis, Stanford University, Stanford, CA.
[8] Kim, H. M., Michelena, N. F., Papalambros, P. Y., and Jiang, T., 2003, "Target Cascading in Optimal System Design," ASME J. Mech. Des., **125**(3), pp. 474–480.
[9] Tosserams, S., 2004, "Analytical Target Cascading: Convergence Improvement by Sub-Problem Post-Optimality Sensitivities," MS thesis, Eindhoven University of Technology, Eindhoven, The Netherlands.
[10] Rodriguez, J. F., Renaud, J. E., and Watson, L. T., 1998, "Convergence of Trust Region Augmented Lagrangian Methods Using Variable Fidelity Approximation Data," Struct. Optim., **15**(3–4), pp. 141–156.
[11] Tosserams, S., Etman, L. F. P., and Rooda, J. E., 2008, "Augmented Lagrangian Coordination for Distributed Optimal Design in MDO," Int. J. Numer. Methods Eng., **73**(13), pp. 1885–1910.
[12] Tosserams, S., Etman, L. F. P., Papalambros, P. Y., and Rooda, J. E., 2006, "An Augmented Lagrangian Relaxation for Analytical Target Cascading Using the Alternating Direction Method of Multipliers," Struct. Multidiscip. Optim., **31**(3), pp. 176–189.
[13] Sobieszczanski-Sobieski, J., and Haftka, R. T., 1997, "Multidisciplinary Aerospace Design Optimization: Survey of Recent Developments," Struct. Optim., **14**(1), pp. 1–23.
[14] Chen, B., Liu, J., Zhong, Y., and Zhou, J., 2000, "Properties of the Coupled Factors in MDO and Their Application in Collaborative Optimization," Proceedings of the DETC'00, ASME 2000 Design Engineering Technical Conferences and Computers and Information in Engineering Conference, Baltimore, MD, Sept. 10–13, Paper No. DETC2000/DAC14516.
[15] Kusiak, A., and Larson, N., 1995, "Decomposition and Representation Methods in Mechanical Design," ASME J. Mech. Des., **117**(B), pp. 17–24.
[16] Michelena, N. F., and Papalambros, P. Y., 1997, "A Hypergraph Framework for Optimal Model-Based Decomposition of Design Problems," Comput. Optim. Appl., **8**(2), pp. 173–196.
[17] Wagner, T. C., 1993, "A General Decomposition Methodology for Optimal System Design," Ph.D. thesis, University of Michigan, Ann Arbor, MI.
[18] Michelena, N. F., Park, H., and Papalambros, P. Y., 2003, "Convergence Properties of Analytical Target Cascading," AIAA J., **41**(5), pp. 897–905.
[19] Tosserams, S., Etman, L. F. P., and Rooda, J. E., 2007, "An Augmented Lagrangian Decomposition Method for Quasiseparable Problems in MDO," Struct. Multidiscip. Optim., **34**(3), pp. 211–227.
[20] Alexandrov, N. M., and Lewis, R. M., 2002, "Analytical and Computational Aspects of Collaborative Optimization for Multidisciplinary Design," AIAA J., **40**(2), pp. 301–309.
[21] Steward, D. V., 1981, *Systems Analysis and Management: Structure, Strategy, Design*, Petrocelli Books, New York.
[22] Steward, D. V., 1965, "Partitioning and Tearing Systems of Equations," SIAM (Soc. Ind. Appl. Math.) J. Numer. Anal., **2**(2), pp. 345–365.
[23] Steward, D. V., 1981, "The Design Structure System: A Method for Managing the Design of Complex Systems," IEEE Trans. Eng. Manage., **EM-28**(3), pp. 71–74.
[24] Browning, T. R., 2001, "Applying the Design Structure Matrix to System Decomposition and Integration Problems: A Review and New Directions," IEEE Trans. Eng. Manage., **48**(3), pp. 292–306.
[25] Sullivan, L. P., 1986, "Quality Function Deployment," Qual. Prog., **19**(6), pp. 39–50.
[26] Suh, N. P., 1990, *The Principles of Design*, Oxford University, New York.
[27] Allison, J. T., Roth, B., Kokkolaras, M., Kroo, I., and Papalambros, P. Y., 2006, "Aircraft Family Design Using Decomposition-Based Methods," Proceedings of the 11th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference, Portsmouth, VA, Sept. 6–8.
[28] Kusiak, A., and Wang, J., 1993, "Decomposition of the Design Process," ASME J. Mech. Des., **115**(4), pp. 687–695.
[29] Michelena, N. F., and Papalambros, P. Y., 1995, "Network Reliability Approach to Optimal Decomposition of Design Problems," ASME J. Mech. Des., **117**(3), pp. 433–440.
[30] Krishnamachari, R. S., and Papalambros, P. Y., 1997, "Optimal Hierarchical Decomposition Synthesis Using Integer Programming," ASME J. Mech. Des., **119**(4), pp. 440–447.
[31] Chen, L., Ding, Z., and Li, S., 2005, "A Formal Two-Phase Method for Decomposition of Complex Design Problems," ASME J. Mech. Des., **127**(2), pp. 184–195.
[32] Rogers, J. L., 1989, "DEMAID—A Design Manager's Aid for Intelligent Decomposition User's Guide," NASA Technical Report No. TM-101575.
[33] Rogers, J. L., 1996, "DEMAID/GA—An Enhanced Design Manager's Aid for Intelligent Decomposition (Genetic Algorithms)," Proceedings of the Sixth AIAA/USAF/NASA/ISSMO Symposium on Multidisciplinary Analysis and Optimization, Sept. 4–6, Paper No. AIAA-96-4157-CP.
[34] Kroo, I., Altus, S., Braun, R., Gage, P., and Sobieski, I., 1994, "Multidisciplinary Optimization Methods for Aircraft Preliminary Design," Proceedings of the Fifth AIAA/USAF/NASA/ISSMO Symposium on Multidisciplinary Analysis and Optimization, Sept. 7–9, Paper No. AIAA-94-4325-CP.
[35] Meier, C., Yassine, A. A., and Browning, T. R., 2007, "Design Process Sequencing With Competent Genetic Algorithms," ASME J. Mech. Des., **129**(6), pp. 566–585.
[36] Altus, S., Kroo, I., and Gage, P., 1996, "Genetic Algorithm for Scheduling and Decomposition of Multidisciplinary Design Problems," ASME J. Mech. Des., **118**(4), pp. 486–489.
[37] Bertsekas, D. P., and Tsitsiklis, J. N., 1997, *Parallel and Distributed Computation: Numerical Methods*, Athena Scientific, Belmont, MA.
[38] Stanton, D., and White, D., 1986, *Constructive Combinatorics*, Springer-Verlag, New York.
[39] Michelena, N. F., and Papalambros, P. Y., 1995, "Optimal Model-Based Decomposition of Powertrain System Design," ASME J. Mech. Des., **117**(4), pp. 499–505.
[40] Chen, L., and Li, S., 2005, "Analysis of Decomposability and Complexity for Design Problems in the Context of Decomposition," ASME J. Mech. Des., **127**(4), pp. 545–557.
[41] Allison, J. T., and Papalambros, P. Y., 2007, "Optimal Partitioning and Coordination Decisions in System Design Using an Evolutionary Algorithm," Proceedings of the Seventh World Conference on Structural and Multidisciplinary Optimization, Seoul, South Korea, May 21–25.
[42] McCleer, P. J., 1995, "Electric Drives for Pump, Fan, and Compressor Loads in Automotive Applications," Proceedings of the IEEE International Symposium on Industrial Electronics, Jul. 10–14, pp. 80–85.
[43] Kassakian, J. G., 1996, "Future of Power Electronics in Advanced Automotive Electrical Systems," IEEE Annual Power Electronics Specialists Conference, pp. 7–14.
[44] Kokkolaras, M., Mourelatos, Z., Louca, L., Filipi, Z., Delagrammatikas, G., Stefanopoulou, A., Papalam-bros, P. Y., and Assanis, D., 2005, "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 Trans, **114**(6), pp. 1651–1660.
[45] 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," Proceedings of the SAE World Congress, Apr. 11–14, Paper No. SAE-2005-01-0016.
[46] Mellor, P. H., Roberts, D., and Turner, D. R., 1991, "Lumped Parameter Thermal Model for Electrical Machines of TEFC Design," IEE Proceedings B: Electric Power Applications, **138**(5), pp. 205–218.
[47] Kenjo, T., and Nagamori, S., 1985, *Permanent-Magnet and Brushless DC Motors*, Oxford University, New York.
[48] Gieras, J. F., and Wing, M., 2002, *Permanent Magnet Motor Technology*, Marcel Dekker, New York.
[49] Tuzson, J., 2000, *Centrifugal Pump Design*, Wiley, New York.
[50] Allison, J. T., 2007, "Design Optimization Model for an Automotive Electric Water Pump," http://ode.engin.umich.edu/repository/TR2007.Allison.pdf
[51] Audet, C., and Dennis, J. E., Jr., 2006, "Mesh Adaptive Direct Search Algorithms for Constrained Optimization," SIAM J. Optim., **17**(1), pp. 188–217.
[52] Jones, D. R., Perttunen, C. D., and Stuckman, B. E., 1993, "Lipschitzian Optimization Without the Lipschitz Constant," J. Optim. Theory Appl., **79**(1), pp. 157–181.
[53] He, J., Watson, L., Ramakrishnan, N., Shaffer, C. A., Verstak, A., Jiang, J., Bae, K., and Tranter, W. H., 2002, "Dynamic Data Structure for a Direct Search Algorithm," Comput. Optim. Appl., **23**, pp. 5–25.