7th World Congress on Structural and Multidisciplinary Optimization
COEX Seoul, 21 May - 25 May 2007, Korea

Optimal Partitioning and Coordination Decisions in System Design Using an Evolutionary Algorithm

James T. Allison, Panos Y. Papalambros

Department of Mechanical Engineering, The University of Michigan
Ann Arbor, Michigan U.S.A.
{optimize,pyp}@umich.edu

1. Abstract
Use of decomposition-based design optimization methods requires a priori selection of system partitioning and of the corresponding coordination strategy. Typically, partitioning systems into smaller, easier to solve subproblems leads to more complicated, computationally expensive coordination strategies. Previous optimal partitioning techniques have not addressed the coordination issue explicitly. A Pareto-optimal problem formulation was proposed recently to address simultaneous partitioning and coordination decisions. Due to the combinatorial nature of this problem, exhaustive enumeration was used to demonstrate the approach to small system examples. This article presents and demonstrates an evolutionary algorithm that can solve this problem for systems of moderate size. A readily partitioned truss design formulation is introduced on which design problems with a wide variety of interaction patterns and system sizes can be based. An eight-bar truss problem is used to demonstrate the effectiveness of the proposed evolutionary algorithm.

2. Keywords: Decomposition-based design optimization, evolutionary algorithm, partitioning, coordination, truss design.

3. Introduction
Decomposition-based design optimization methods can ease difficulties associated with complex system design. Their application, however, requires that both a system partition and a coordination strategy are defined a priori. The partitioning task involves clustering m analysis functions required for the system design problem into N subproblems. Subproblem solution must be coordinated in a way that leads to a consistent and optimal system design. Partitioning and coordination decisions should be made such that the decomposed problem is less complex to solve than the original undecomposed problem. Engineering insight traditionally is used to partition a system, and system designers can select a coordination strategy based on their experience or follow qualitative selection guidelines available in the literature. Formal approaches can lead to improved partitioning and coordination decisions [1]. System partitioning methods have been developed that cluster system elements into balanced subproblems with minimal communication requirements between subproblems [2, 3, 4, 5, 6]. The execution sequence of subproblems is one aspect of coordination that has been studied thoroughly. Design task sequencing, a problem analogous to subproblem sequencing, has been solved using heuristic rules [7, 8] and genetic algorithms [3, 9, 10] in order to reduce design task iterations.

Partitioning and coordination decisions have been treated largely as independent, thus ignoring potential interaction and synergy between these decisions. Kusiak and Wang did account for part of this interaction using a sequential approach where subproblem order is considered after partitioning [1]. Altus, Kroo, and Gage employed a genetic algorithm to partition a system and simultaneously determine the evaluation order of analysis functions within subproblems [3]. Since only a single solution was presented, the tradeoff between subproblem and coordination difficulty could not be evaluated. This tradeoff exists because finely partitioned systems with easily solved subproblems have numerous interconnections and a complex coordination problem, while coarsely partitioned systems are generally easy to coordinate, but have larger and more difficult to solve subproblems. This tradeoff was studied recently for three small example systems using exhaustive enumeration [11]. In addition to providing quantitative guidance for making Pareto-optimal partitioning and coordination decisions, tradeoff data can be used to assess the sensitivity of a system to partitioning and its suitability for decomposition-based design optimization. It was also shown that partitioning and coordination decisions are coupled, and a simultaneous decision-making approach is required to identify Pareto-optimal decisions. The analysis was limited to small systems (less than seven analysis functions) because exhaustive enumeration was employed as the solution algorithm. This article demonstrates how to identify Pareto-optimal decisions and obtain tradeoff data for systems of practical size using an evolutionary algorithm.

3.1. System Representation

Solution of a system design problem requires the evaluation of a coupled set of analysis functions that generate design objective and constraint values for a particular design. The i-th analysis function $\mathbf{a}_i(\mathbf{x}_i, \mathbf{y}_i)$ depends on $\mathbf{x}_i$, a subset of the vector of all system design variables $\mathbf{x}$; and $\mathbf{y}_i$, a subset of the vector of all analysis coupling variables $\mathbf{y}$. Analysis coupling variables are analysis outputs that are required as inputs to other analysis functions, and $\mathbf{y}_i$ is the set of analysis coupling variables input to the i-th analysis function.

Graphs have been used to represent the structure of a system design problem. The design structure matrix (DSM) is the adjacency matrix of a directed graph that can represent interactions between analysis functions [7]. Each vertex corresponds to an analysis function, and arc $\langle i, j\rangle$ exists if $\mathbf{a}_j(\mathbf{x}_j, \mathbf{y}_j)$ depends on the output of $\mathbf{a}_i(\mathbf{x}_i, \mathbf{y}_i)$. The specific coupling variable $\mathbf{y}_{ij}$ is the quantity passed from $\mathbf{a}_j(\mathbf{x}_j, \mathbf{y}_j)$ to $\mathbf{a}_i(\mathbf{x}_i, \mathbf{y}_i)$; $\mathbf{y}_{ij}$ is a component of $\mathbf{y}_i$ and exists if element ij of the DSM is nonzero. The DSM can help guide sequencing decisions. The functional dependence table (FDT) represents functional dependence on design and coupling variables [12]. It can be viewed as the incidence matrix for a hypergraph where design objective and constraint functions are vertices and hyperedges are design or coupling variables [4]. This representation is undirected, and so the FDT cannot be used to make subproblem sequencing decisions, but it is useful for partitioning decisions since it accounts for dependence on both design and coupling variables.

A directed graph can be defined that contains all information in both the FDT and DSM. Figure 1 shows a directed graph representation of an example system where arcs indicate analysis function dependence on design and coupling variables [11]. The analysis structure of a system with n design variables and m analysis functions can be compactly represented using the corresponding $(n+m)\times(n+m)$ adjacency matrix. Since design variables are independent, all adjacency matrix rows corresponding to design variables are zero, and no information is lost by omitting these rows. The reduced adjacency matrix for the example system is given in Fig. 2.

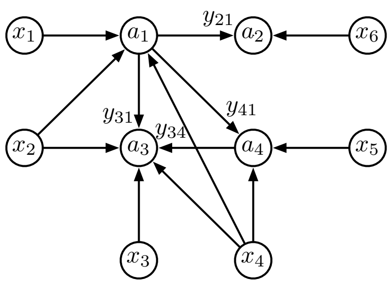

Figure 1: Directed graph representation of analysis structure (after [11])

$\mathbf{A} =$

	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

Figure 2: Reduced adjacency matrix representation

3.2. Simultaneous Partitioning and Coordination Decisions

The reduced adjacency matrix $\mathbf{A}$ is a compact representation of the combined information required to make simultaneous partitioning and coordination decisions. It is the basis of the decision model introduced in [11], and used also here. Given a system partition and subproblem sequence, the coordination problem size (CS) and subproblem sizes ($SS_i,\ i \in \{1, 2, \dots, N\}$) can be computed using $\mathbf{A}$. These size metrics are proxies for the computational expense associated with solving the coordination problem and subproblems, respectively. CS is the number of consistency constraints that must be solved in the coordination problem and SS_i is the size of the i-th subproblem.

A restricted growth string (RGS) [13] $\mathbf{p}$ of length m is used to specify a system partition. Analysis function i belongs to the subproblem identified by the value of p_i. Redundant representations of partitions are avoided since, as an RGS, $\mathbf{p}$ must satisfy [13]:

$$p_1 = 1 \ \wedge \ p_i \leq \max\{p_1, p_2, \dots, p_{i-1}\} + 1 \tag{1}$$

The coordination decision model used here is restricted to subproblem sequence choice. A more sophisticated model that also includes consistency constraint allocation is under development. Subproblem sequence is represented by $\mathbf{o}_s$, a vector of length N where the value of o_{si} is the evaluation position of subproblem i, and $o_{si} \neq o_{sj},\ \forall\, i, j \in \{1, 2, \dots, N\}$.

A simultaneous approach to partitioning and coordination decision-making considers $\mathbf{p}$ and $\mathbf{o}_s$ together, rather than independently or in sequence, and can be formulated as a multiobjective optimization problem:

$$\min_{\mathbf{p}, \mathbf{o}_s} \quad \{CS, SS_{\max}\}, \tag{2}$$

where $SS_{\max}$ is the maximum subproblem size. Since specification of $\mathbf{p}$ and $\mathbf{o}_s$ are coupled tasks, a simultaneous approach is required to obtain Pareto-optimal solutions to Eq. (2).

4. Evolutionary Algorithm

Evolutionary algorithms (EAs) seek to improve the 'fitness' of a population composed of candidate problem solutions through two primary means: selection and variation [14]. A subset of the population is chosen to produce the population for the next generation using variation operators. This process is repeated until some termination criterion is met. EAs are particularly effective at searching very large decision spaces and arriving at good, even if not optimal, solutions in a reasonable amount of time. EAs handle multiobjective problems with little additional expense. The discrete decision space of Eq. (2) is vast; the number of possible partitioning and coordination instances increases exponentially with m. The partitioning and sequencing problems are themselves NP-complete and NP-hard, respectively. An exact solution for the combined problem is possible only for very small systems. It is proposed here that an EA tailored to the properties of the simultaneous partitioning and coordination problem is an effective solution technique for systems of practical size.

In an EA each individual is abstractly represented by its chromosome, or genotype. The genotype representation must be compatible with the variation operators used in the EA. Frequently the representation of a candidate solution in the original problem statement, also called the phenotype, is not suitable for use as a genotype representation. In this case an appropriate genotype space must be defined along with a surjective mapping onto the phenotype space. Rather than devise variation operators that apply directly to $\mathbf{p}$ and $\mathbf{o}_s$, a genotype compatible with standard crossover and mutation variation operators was developed.

Choice of genotype representation can strongly influence EA success. An ideal representation should not increase problem difficulty, should enable variation operators to work properly, and should result in a process that is robust to solution location [15]. To meet these requirements it is important that a representation have good locality (i.e., small changes in the genotype space result in small changes in the phenotype space) and little bias toward particular genotypes.

4.1. Partition Genotype Representation

Rather than attempt to define an effective variation operator to operate directly on restricted growth strings, the system partition is represented in the genotype space using $\hat{\mathbf{p}}$, where $\hat{p}_i \in \{1, 2, \dots, m\},\ i = 1, 2, \dots, m$. Note that the number of analysis functions m is the maximum possible number of subproblems N. The vector $\hat{\mathbf{p}}$ defines a partition, although not uniquely. There exist m^m possible ways to assign values to $\hat{\mathbf{p}}$, while the number of unique partitions is the m-th Bell number B_m [13]. The ratio $\mu = m^m/B_m$ quantifies redundancy incurred by using $\hat{\mathbf{p}}$, and increases quickly with m. For a system of size $m = 6$, $\mu = 229.83$.

An algorithm was developed that maps $\hat{\mathbf{p}}$ values in the genotype space to $\mathbf{p}$ values in the phenotype space. Figure 3 illustrates (for $m = 6$) the normalized frequency of partition sizes using the phenotype and genotype representations, where N is the partition size. Both distributions are biased toward intermediate values. While genotype bias impedes EA effectiveness, this representation was selected for its favorable properties under crossover and mutation.

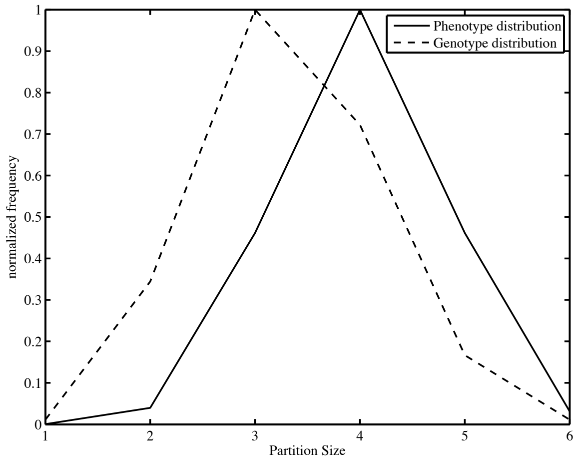

Figure 3: Genotype and Phenotype representation partition size distributions

4.2. Sequence Genotype Representation

The subproblem sequence representation poses a challenge because the length of $\mathbf{o}_s$ depends on $\mathbf{p}$. An extension of the Random Key (RK) representation can be utilized to address this problem [16]. A random key is a real-valued vector that can be used to encode an integer sequence. For example, suppose $\hat{\mathbf{o}}_s$ is a real valued vector of length N where $0 \leq \hat{o}_{si} \leq 1,\ i = 1, 2, \ldots, N$. The components of $\hat{\mathbf{o}}_s$ are then sorted in ascending order, and the order of the original component indices after sorting defines the sequence. RK representations have proven to be more effective than using variation operators designed for permutations. RKs exhibit high locality, and standard variation operators for real values are effective [15].

RK representation works well when the number of elements to be sequenced is fixed. Since this is not the case here, an extension was made. The vector $\hat{\mathbf{o}}$ contains an element for each analysis function: $0 \leq \hat{o}_i \leq 1,\ i = 1, 2, \ldots, m$. The meaning of $\hat{\mathbf{o}}$ depends on $\mathbf{p}$. If $\mathcal{P}_j$ is the set of analysis function indices that belong to the j-th subproblem, then $\hat{o}_{sj} = \sum_{i \in \mathcal{P}_j} \hat{o}_i / |\mathcal{P}_j|$. The sequence $\mathbf{o}_s$ is then obtained through sorting $\hat{\mathbf{o}}_s$.

The number of possible subproblem sequences increases with partition size, biasing distribution of candidate subproblem sequences toward finer partitions (Fig. 4). All possible pairs of $\mathbf{p}$ and $\mathbf{o}_s$ for a given system comprise its phenotype space. The distribution of all these instances for a system of size $m = 6$ is shown in Figure 5. The bias present here means that an EA will likely expend most of its effort exploring candidate solutions with 4, 5, or 6 subproblems.

5. Comparative Examples

The exact Pareto-optimal solutions for three small example systems were presented in [11] and are compared here against results using the EA described above. The first two example systems are defined by the adjacency matrices $\mathbf{A}_1$ and $\mathbf{A}_2$:

$$\mathbf{A}_1 = \begin{bmatrix} 0&1&1&1&0&1&0&1&1&1&0&1 \\ 0&0&1&1&1&0&1&1&0&1&1&1 \\ 1&0&0&1&1&1&1&1&1&1&1&1 \\ 1&1&1&0&0&1&1&1&1&1&1&1 \\ 1&1&0&1&0&0&1&1&1&0&0&1 \end{bmatrix}, \ \mathbf{A}_2 = \begin{bmatrix} 0&1&1&1&0&0&1&1&1&1&1&1&1&1&0&1 \\ 1&0&1&1&1&0&0&1&1&1&0&1&0&1&1&0 \\ 1&1&0&1&0&1&0&1&0&1&1&1&1&0&1&0 \\ 1&1&1&0&0&0&1&0&1&0&0&1&0&0&0&0 \\ 1&1&1&1&0&0&1&1&1&1&1&1&1&0&1&0 \\ 1&1&1&1&1&0&1&1&1&0&1&1&1&0&1&1 \end{bmatrix}.$$

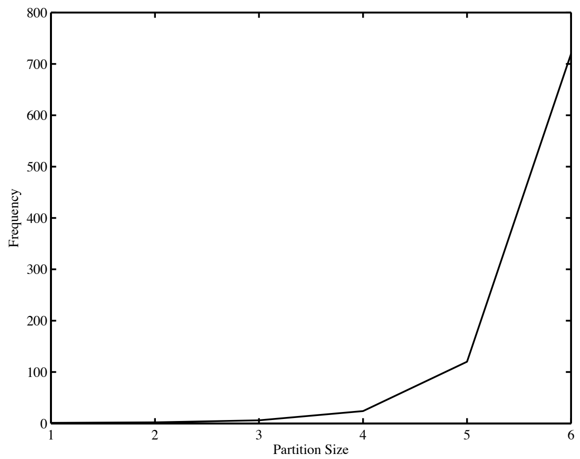

Figure 4: Subproblem sequence distribution

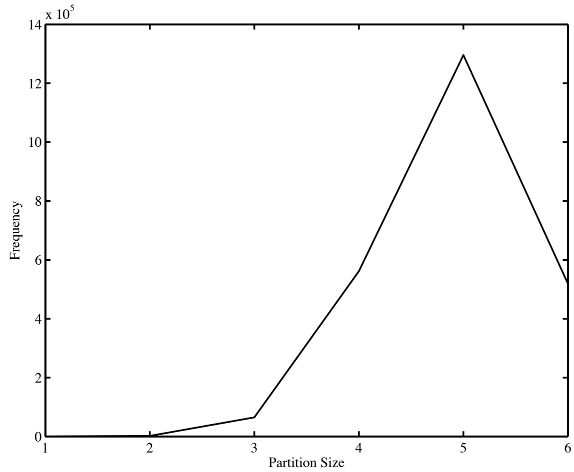

Figure 5: Combined subproblem sequence and partition size distribution

The set of non-dominated points in the objective space identified by the EA was recorded for each system. Figures 6 and 7 compare these points against the known Pareto points. In System 1 the EA failed to identify two Pareto points, and two of non-dominated points were not Pareto points. In System 2 three Pareto points were not identified. It appears that the EA has difficulty identifying Pareto-optimal solutions with small $SS_{\max}$ (i.e., large partition sizes). This is unexpected given the representation bias, and is a topic of continued work.

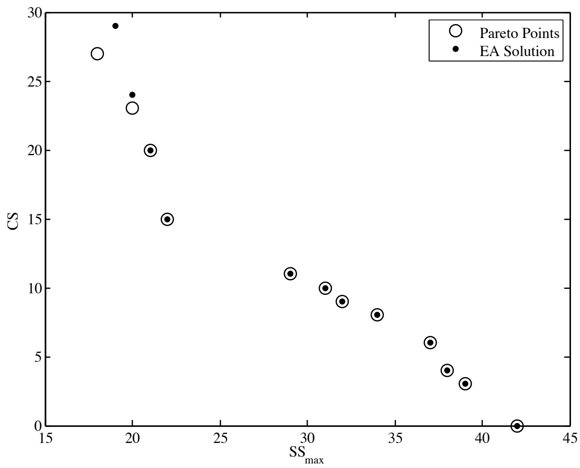

Figure 6: EA results for first example system

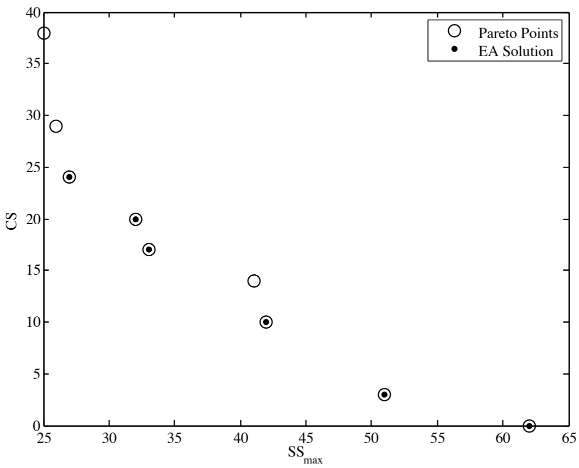

Figure 7: EA results for second example system

The third example system from [11] corresponds to the design of an electric water pump for use in an automotive cooling system, and its analysis structure is defined by $\mathbf{A}_3$. Figure 8 illustrates that for this smaller system ($m = 4$), the EA successfully identified all four Pareto-optimal solutions. The following section develops a physically meaningful test problem formulation that can represent systems of arbitrary size with a variety of possible analysis structures. The EA is then applied to a test problem too large for exhaustive enumeration.

6. Decomposition-based Truss Design

Consider a family of structural trusses where all members are secured via pin joints, two or more ground joints are fixed, at least one load is applied to a non-ground joint, and truss topology may or may not exhibit static indeterminacy. Figure 9 illustrates one such truss with five joints and six members. Joints

$$\mathbf{A}_3 = \begin{bmatrix} 0 & 1 & 1 & 0 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 \end{bmatrix}$$

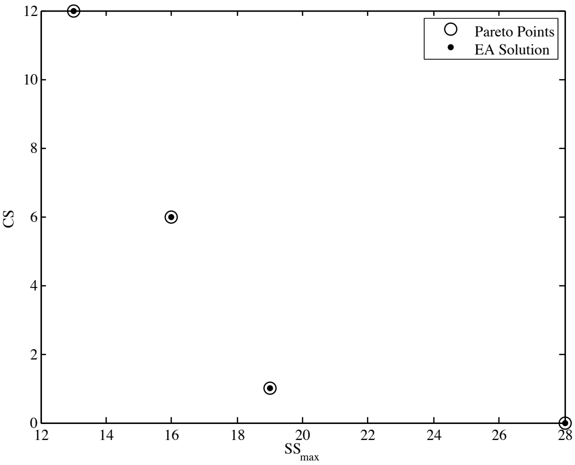

Figure 8: EA results for third example system

1 and 4 are fixed, and joints 2 and 3 support external loads. A free-body diagram for member $\{2,4\}$ is provided.

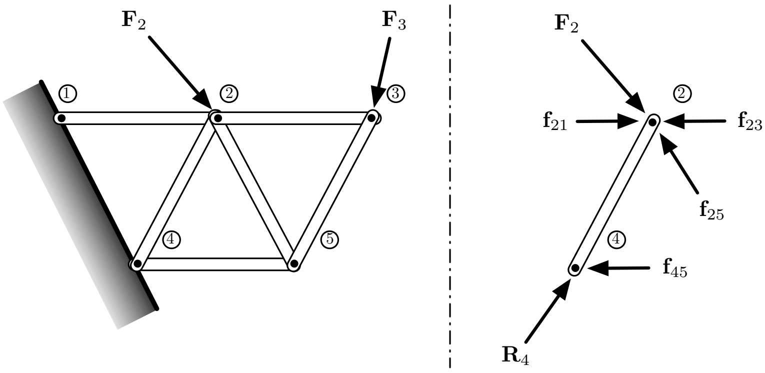

Figure 9: Truss geometry and free-body diagram

Here $\mathbf{F}_i$ is the external force applied to joint i, $\mathbf{R}_i$ is the reaction force at joint i, $\mathbf{u}_i$ is the two-dimensional position for joint i before deformation, and $\mathbf{d}_i$ is the position after deformation. All members have circular cross section, and the radius of member $\{i,j\}$ is r_{ij}. The internal force of member $\{i,j\}$ in scalar form is f_{ij}, and the vector form is given by:

$$\mathbf{f}_{ij} = f_{ij}\frac{\mathbf{d}_j - \mathbf{d}_i}{\|\mathbf{u}_j - \mathbf{u}_i\|_2}, \tag{3}$$

which can also be interpreted as the force exerted on joint i by member $\{i,j\}$. The force exerted by this same member on joint j is $\mathbf{f}_{ji} = -\mathbf{f}_{ij}$. Note that the member forces depend on deformed joint locations, accounting for one source of nonlinearity. The set of all joint indices is $\mathcal{J}$ and the set of all unordered member index pairs is $\mathcal{M}$. The ordered pair $\{\mathcal{J},\mathcal{M}\}$ comprises an undirected graph that describes truss topology. The set of joints connected to joint i is $\mathcal{A}_i = \{k|\{i,k\} \in \mathcal{M}\}$. The indices of all fixed ground joints comprise the set $\mathcal{G}$, and $\mathcal{L}$ is the set of all joint indices with an applied external load force.

The truss design problem is to select the radii of all members $\mathbf{r}$ and positions of all movable joints $\mathbf{m}$ (where $\mathbf{m} = [\mathbf{u}_{i_1}, \mathbf{u}_{i_1}, \ldots, \mathbf{u}_{i_k}]$, $\{i_1, i_2, \ldots, i_k\} = \mathcal{J}\backslash(\mathcal{G} \cup \mathcal{L})$, and $k = |\mathcal{J}\backslash(\mathcal{G} \cup \mathcal{L})|$) such that the system mass is minimized without violating stress or buckling constraints. Ground and load joints have prescribed locations in the design problem, but the other joints are considered moveable, i.e., their undeformed location is at the discretion of the designer. Since statically indeterminate systems are allowed, both structural compatibility and joint equilibrium equations are included in the analysis. State variables include internal member forces ($\mathbf{f} = [f_{i_1 j_1}, f_{i_2 j_2}, \ldots, f_{i_k j_k}]$, $\{\{i_1, j_1\}, \{i_2, j_2\}, \ldots, \{i_k, j_k\}\} = \mathcal{M}$, and k is the number of truss members), the deformed positions of non-ground joints ($\tilde{\mathbf{d}} = [\mathbf{d}_{i_1}, \mathbf{d}_{i_2}, \ldots, \mathbf{d}_{i_k}]$, $\{i_1, i_2, \ldots, i_k\} = \mathcal{J} \backslash \mathcal{G}$, and $k = |\mathcal{J} \backslash \mathcal{G}|$), and the reaction forces ($\mathbf{R} = [\mathbf{R}_{i_1}, \mathbf{R}_{i_2}, \ldots, \mathbf{R}_{i_{|\mathcal{G}|}}]$, $\{i_1, i_2, \ldots, i_{|\mathcal{G}|}\} = \mathcal{G}$). The all-at-once (AAO) optimization formulation [17] for the general truss design problem includes both design ($\mathbf{m}, \mathbf{r}$) and state ($\mathbf{f}, \tilde{\mathbf{d}}, \mathbf{R}$) variables as decision variables, and treats state equations as equality constraints:

$$
\begin{aligned}
\min_{\mathbf{m},\mathbf{r},\mathbf{f},\tilde{\mathbf{d}},\mathbf{R}} \quad & \sum_{\{i,j\} \in \mathcal{M}} \Omega_{ij} \qquad (4) \\
\text{subject to:} \quad & |\sigma_{ij}| - \sigma_{\text{allow}} \leq 0, \quad \forall \{i,j\} \in \mathcal{M} \\
& -f_{ij} - b_{ij} \leq 0, \quad \forall \{i,j\} \in \mathcal{M} \\
& \|\mathbf{d}_i - \mathbf{d}_j\|_2 - \|\mathbf{u}_i - \mathbf{u}_j\|_2 - \frac{f_{ij} \|\mathbf{u}_i - \mathbf{u}_j\|_2}{\pi r_{ij}^2 E} = 0, \quad \forall \{i,j\} \in \mathcal{M} \\
& \sum_{k \in \mathcal{A}_i} \mathbf{f}_{ik} + \mathbf{F}_i + \mathbf{R}_i = \mathbf{0}, \quad \forall i \in \mathcal{J} \\
\text{where:} \quad & \sigma_{ij} = \frac{f_{ij}}{\pi r_{ij}^2}, \; b_{ij} = \frac{\pi^3 r_{ij}^4 E}{4 \|\mathbf{u}_i - \mathbf{u}_j\|_2^2}, \; \Omega_{ij} = \rho \pi r_{ij}^2 \|\mathbf{u}_i - \mathbf{u}_j\|_2
\end{aligned}
$$

Design parameters include material density ρ, elastic modulus E, allowable stress σ_{allow}, fixed ground and load joint positions ($\mathbf{d}_i, \forall i \in \mathcal{G} \cup \mathcal{L}$), and applied loads ($\mathbf{F}_i, \forall i \in \mathcal{L}$). The mass of member $\{i, j\}$ is Ω_{ij} and the Euler buckling load for member $\{i, j\}$ is b_{ij}.

The formulation presented above is readily partitioned for use with decomposition-based design optimization. An analysis function is defined for each truss member that computes the internal force, mass, stress, buckling criteria and state equation residuals for its member:

$$
[f_{ij}, \Omega_{ij}, \sigma_{ij}, b_{ij}, \boldsymbol{\Delta_{ij}}] = \mathbf{a}_{q(i,j)}(r_{ij}, \mathbf{u}_i, \mathbf{u}_j, \mathbf{d}_i, \mathbf{d}_j, \mathbf{R}_i, \mathbf{R}_j, \mathbf{f}_i^{ij}, \mathbf{f}_j^{ij}) \qquad (5)
$$

The vector $\boldsymbol{\Delta_{ij}}$ contains the three residual values for the structural compatibility and joint equilibrium state equations, which are constrained to be zero in Eq. (4). The function $q(i, j)$ maps the joint indices for member $\{i, j\}$ to the index of the analysis function that computes responses for that member. The two-dimensional vector $\mathbf{f}_i^{ij}$ is the cumulative force from adjacent members acting on member $\{i, j\}$:

$$
\mathbf{f}_i^{ij} = \sum_{\{i,k\} \in \mathcal{A}_i \backslash \{i,j\}} \mathbf{f}_{ik} \qquad (6)
$$

The truss member analysis functions follow the form $\mathbf{a}_k(\mathbf{x}_k, \mathbf{y}_k)$ introduced earlier, where $k = q(i, j)$, $\mathbf{x}_k = [r_{ij}, \mathbf{u}_i, \mathbf{u}_j, \mathbf{d}_i, \mathbf{d}_j, \mathbf{R}_i, \mathbf{R}_j]$, and $\mathbf{y}_k = [\mathbf{f}_i^{ij}, \mathbf{f}_j^{ij}]$. The coupling variables are formed using member force values and geometry information (Eqs. (3) and (6)). Since f_{ij} is the only analysis output required by other analysis functions, it is the only coupling variable, and is of prime interest when making partitioning and coordination decisions for the partitioned truss design problem. All other analysis outputs are local quantities.

These analysis functions can be clustered to form subproblems. When members in different subproblems are connected at common joints, the corresponding internal member forces are coupling variables between the subproblems. In addition, undeformed positions of common joints are shared design variables. Deformed locations and reaction forces for common joints are also shared variables since the state variables $\mathbf{d}_i$ and $\mathbf{R}_i$ are treated as design variables. The objective function is additively separable, enabling the formation of local subproblem objective functions that consist of the mass of all members in a subproblem.

A wide variety of analysis structures and system sizes are available using this formulation depending on truss size and topology, making Eq. (4) an excellent platform for testing the performance of decomposition-based design optimization methods, as well as methods for combined partition and coordination decision-making.

7. Example: Eight-bar Truss

An eight-bar truss problem with topology adopted from [18] was formulated and solved for use in demonstrating the evolutionary algorithm on a system too large for an exhaustive enumeration approach. This truss is illustrated in Figure 10.

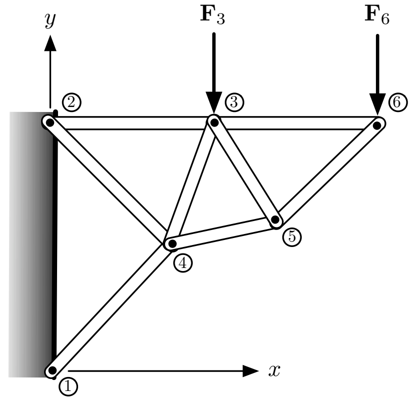

Figure 10: Geometry and applied loads for the 8-bar truss problem

The member radii values are $\mathbf{r} = [r_{14}, r_{24}, r_{23}, r_{34}, r_{45}, r_{35}, r_{36}, r_{56}]$, the movable joint positions are $\mathbf{m} = [\mathbf{u}_4, \mathbf{u}_5]$, the deformed positions of non-ground joints are $\tilde{\mathbf{d}} = [\mathbf{d}_3, \mathbf{d}_4, \mathbf{d}_5, \mathbf{d}_6]$, and the reaction forces are $\mathbf{R} = [\mathbf{R}_1, \mathbf{R}_2]$. The reduced adjacency matrix for this system design problem is:

$$\mathbf{A}_4 = \begin{array}{r|cccccccc cccccccc cc cccc cc}
 & \multicolumn{8}{c}{\overbrace{\hphantom{0\ 0\ 0\ 0\ 0\ 0\ 0\ 0}}^{\mathbf{a}}} & \multicolumn{8}{c}{\overbrace{\hphantom{0\ 0\ 0\ 0\ 0\ 0\ 0\ 0}}^{\mathbf{r}}} & \multicolumn{2}{c}{\overbrace{\hphantom{0\ 0}}^{\mathbf{m}}} & \multicolumn{4}{c}{\overbrace{\hphantom{0\ 0\ 0\ 0}}^{\tilde{\mathbf{d}}}} & \multicolumn{2}{c}{\overbrace{\hphantom{0\ 0}}^{\mathbf{R}}} \\
\hline
\mathbf{a}_{q(1,4)} & 0 & 1 & 0 & 1 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 & 0 \\
\mathbf{a}_{q(2,4)} & 1 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 1 \\
\mathbf{a}_{q(2,3)} & 0 & 1 & 0 & 1 & 0 & 1 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 \\
\mathbf{a}_{q(3,4)} & 1 & 1 & 1 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 1 & 0 & 0 & 0 & 0 \\
\mathbf{a}_{q(4,5)} & 1 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 1 & 0 & 1 & 1 & 0 & 0 & 0 \\
\mathbf{a}_{q(3,5)} & 0 & 0 & 1 & 1 & 1 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 1 & 0 & 1 & 0 & 0 & 0 \\
\mathbf{a}_{q(3,6)} & 0 & 0 & 1 & 1 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 \\
\mathbf{a}_{q(5,6)} & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 1 & 1 & 0 & 0
\end{array}$$

Observe that the submatrix formed by the first $m = 8$ rows and columns of $\mathbf{A}_4$ is symmetric. This is true for any system defined using Eq. (4) since internal member forces are the only coupling variables. Even with this limitation a wide variety of interesting system interaction patterns can be studied.

The design parameters used in this problem and the optimal geometry are given in Table 1. The optimal mass is 1.80 kg; as expected, the stress constraints for the members in tension ($\{2,3\}, \{2,4\}, \{3,5\}, \{3,6\}$) are active, and the buckling constraints for the members in compression ($\{1,4\}, \{3,4\}, \{4,5\}, \{5,6\}$) are active.

Table 1: Design parameters and optimal geometry for the 8-bar truss problem

Design Parameters:				Optimal Geometry:			
ρ	$7.80 \cdot 10^3$ kg/m^3	$\mathbf{d}_3$	$[300, 300]$ mm	r_{14}	3.44 mm	r_{36}	1.33 mm
E	200 GPa	$\mathbf{d}_6$	$[600, 300]$ mm	r_{23}	1.70 mm	r_{45}	2.74 mm
σ_{allow}	250 MPa	$\mathbf{F}_3$	$[0, -1000]$ N	r_{24}	1.10 mm	r_{56}	2.61 mm
$\mathbf{d}_1$	$[0, 0]$ mm	$\mathbf{F}_6$	$[0, -1000]$ N	r_{34}	2.50 mm	$\mathbf{u}_4$	$[232, 108]$ mm
$\mathbf{d}_2$	$[0, 300]$ mm			r_{35}	0.83 mm	$\mathbf{u}_5$	$[434, 180]$ mm

The EA was used to solve Eq. (2) with the analysis structure defined by $\mathbf{A}_4$, and the resulting non-dominated solutions are displayed in Fig. 11. The exact solution is unavailable due to system size, so the number of actual Pareto-optimal solutions identified is unknown. The EA parameters were adjusted until the same best set of non-dominated solutions was generated consistently over several runs. The approximate Pareto set illustrates the CS–$SS_{\max}$ tradeoff for this system and indicates that this problem is a good candidate for partitioning since $SS_{\max}$ can be reduced by almost half before incurring much coordination expense.

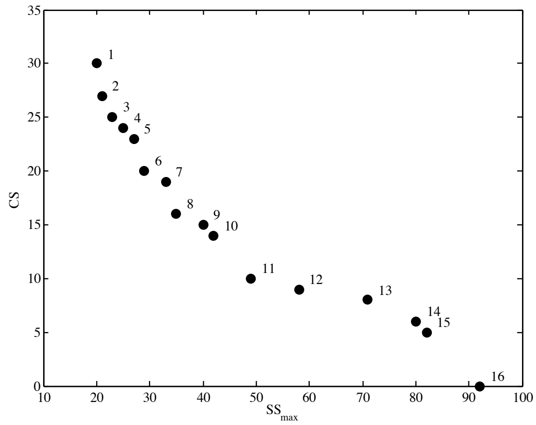

Figure 11: Non-dominated solutions for 8-bar truss problem

Point 16 corresponds to a partition with only one subproblem ($\mathbf{p} = [1, 1, 1, 1, 1, 1, 1, 1]$, $\mathbf{o}_s = [1]$) and has a large subproblem size ($SS = 92$) but no coordination expense. The solution approach represented by point 16 is equivalent to solving Eq. (4) directly without decomposition. Moving from point 16 to point 11 ($\mathbf{p} = [1, 1, 2, 1, 1, 2, 2, 2]$, $\mathbf{o}_s = [2, 1]$) requires dividing the analysis functions into two subproblems and increases the coordination problem size to 10, but reduces $SS_{\max}$ from 92 to 49. Moving from point 11 to point 6 ($\mathbf{p} = [1, 2, 2, 1, 3, 4, 4, 4]$, $\mathbf{o}_s = [3, 4, 1, 2]$) also increases CS by 10, but only reduces $SS_{\max}$ by 20. Point 11 appears to be an appropriate choice since moving away from it leads to a sharp increase in either CS or $SS_{\max}$.

Point 1 ($\mathbf{p} = [1, 2, 3, 4, 5, 6, 7, 7]$, $\mathbf{o}_s = [3, 5, 4, 2, 6, 1, 7]$) has a partition size of $N = 7$ and is the finest partition selected by the EA. Either reducing $SS_{\max}$ below 20 is unachievable by choosing $\mathbf{p} = [1, 2, 3, 4, 5, 6, 7, 8]$, or the EA failed to identify a Pareto-optimal solution with a partition size of $N = 8$. The latter possibility is tenable given that the EA had difficulty identifying low $SS_{\max}$ solutions in the comparative examples.

Although the EA cannot generate exact solutions, it provides valuable information for assessing the suitability of a system for decomposition-based design optimization and for making partitioning and coordination decisions. The sensitivity of a system design problem to increased partition size can be visualized using CS–$SS_{\max}$ tradeoff data, and the EA efficiently identifies (approximate) Pareto-optimal solutions.

8. Concluding Remarks

An evolutionary algorithm was developed for the solution of the combined partitioning and coordination decision problem. The results from this algorithm were compared against exact solutions for small systems, demonstrating that a good approximation to Pareto-optimal solutions can be obtained using the EA. A formulation for a truss structure design with arbitrary size and topology was introduced as a test example. The EA successfully generated a set of approximate Pareto-optimal solutions for a truss design of moderate size but with high computational complexity. Continuing work involves improved efficiency of the EA and the inclusion of more sophisticated coordination decision models.

9. 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. The authors would also like to thank Michael Kokkolaras for reviewing the article and offering helpful suggestions.

10. References

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

[2] N. F. Michelena, T. Jiang, and P. Y. Papalambros. Decomposition of simultaneous analysis and design models. In *Proceedings of the 1995 1st World Congress of Structural and Multidisciplinary Optimization, WCSMO-1, May 28-Jun 2 1995*, May 28-June 2, 1995 1995.

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

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

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

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

[7] D. V. Steward. The design structure system: a method for managing the design of complex systems. *IEEE Transactions on Engineering Management*, EM-28(3):71–4, 08/ 1981.

[8] J.L. Rogers. DeMAID—a design manager's aid for intelligent decomposition user's guide. Technical Report TM-101575, NASA, 1989.

[9] J. L. Rogers. DeMAID/GA - an enhanced design manager's aid for intelligent decomposition (genetic algorithms). In *6th AIAA/USAF/NASA/ISSMO Symposium on Multidisciplinary Analysis and Optimization, Seattle, WA*, September 4-6, 1996.

[10] C. Meier, A. A. Yassine, and T. R. Browning. Design process sequencing with competent genetic algorithms. *To appear in Journal of Mechanical Design, Transactions of the ASME*, 2007.

[11] J. T. Allison, M. Kokkolaras, and P. Y. Papalambros. Optimal partitioning and coordination decisions in decomposition-based design optimization. In *ASME Design Engineering Technical Conference DETC2007-34698*, Sept. 4-Sept 7 2007 (in review).

[12] T.C. Wagner. *A General Decomposition Methodology For Optimal System Design*. PhD thesis, University of Michigan, 1993.

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

[14] A. E. Eiben and J. E. Smith. *Introduction to Evolutionary Computing*. Springer-Verlag, Germany, 2003.

[15] F. Rothlauf. *Representations for Genetic and Evolutionary Algorithms*. Springer-Verlag, Berlin, second edition, 2006.

[16] J. C. Bean. Genetic algorithms and random keys for sequencing and optimization. *ORSA Journal on Computing*, 6:154–60, 1994.

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

[18] M. Giger and P. Ermanni. Evolutionary truss topology optimization using a graph-based parameterization concept. *Structural and Multidisciplinary Optimization*, 32(4):313–326, 2006.