Computational Optimization and Applications, 18, 273–293, 2001

Convergence Criteria for Hierarchical Overlapping Coordination of Linearly Constrained Convex Design Problems

HYUNGJU PARK park@oakland.edu
Department of Mathematics and Statistics, Oakland University, Rochester, MI 48309

NESTOR MICHELENA nestorm@umich.edu
Department of Mechanical Engineering, University of Michigan, Ann Arbor, MI 48109

DEVADATTA KULKARNI kulkarni@oakland.edu
Department of Mathematics and Statistics, Oakland University, Rochester, MI 48309

PANOS PAPALAMBROS pyp@umich.edu
Department of Mechanical Engineering, University of Michigan, Ann Arbor, MI 48109

Received April 30, 1998; Accepted July 20, 1999

Abstract. Decomposition of multidisciplinary engineering system design problems into smaller subproblems is desirable because it enhances robustness and understanding of the numerical results. Moreover, subproblems can be solved in parallel using the optimization technique most suitable for the underlying mathematical form of the subproblem. Hierarchical overlapping coordination (HOC) is an interesting strategy for solving decomposed problems. It simultaneously uses two or more design problem decompositions, each of them associated with different partitions of the design variables and constraints. Coordination is achieved by the exchange of information between decompositions. This article presents the HOC algorithm and several new sufficient conditions for convergence of the algorithm to the optimum in the case of convex problems with linear constraints. One of these equivalent conditions involves the rank of the constraint matrix that is computationally efficient to verify. Computational results obtained by applying the HOC algorithm to quadratic programming problems of various sizes are included for illustration.

Keywords: decomposition methods, large-scale optimization, distributed computing, hierarchical coordination

1. Introduction

Engineering design can be viewed as a decision-making process that uses mathematical models to predict design behavior and to select a design whose value is considered satisfactory. A typical approach consists of formulating a design optimization problem using models to estimate design criteria and constraint functions, and applying formal methods to search the design space for an optimum.

In this article, we assume that a design problem can be formulated as a convex optimization problem of the form:

$$\text{find } x \in \mathbb{R}^n \text{ such that } \mathbf{h}(\mathbf{x}) = 0,\ \mathbf{g}(\mathbf{x}) \leq 0 \text{ and } f(\mathbf{x}) \text{ is minimized,}$$

where $f : \mathbb{R}^n \to \mathbb{R}$ and $g_i : \mathbb{R}^n \to \mathbb{R}$ are convex functions, and $h_i : \mathbb{R}^n \to \mathbb{R}$ are affine functions. We assume that the problem above has a nonempty solution set, and that f and g_i are differentiable functions on $\mathbb{R}^n$. Although most design problems are nonlinear, nonconvex problems, many optimization algorithms solve a sequence of approximation problems similar to those considered in this article to arrive at the solution of the original design problem.

In the case of a large nonlinear design problem that involves a significant number of variables and constraints, decomposition of the design problem into smaller design subproblems may be desirable. The subproblems can then be solved in parallel, using the optimization technique most suitable for the underlying submodel, gaining in robustness and interpretation of results, and occasionally also in speed of execution. Moreover, system design problems typically involve several disciplines. Subsystem design teams represent an explicit problem decomposition. Thus, coordinated solution of design subproblems may be the only way to address the overall system problem in a practical and robust manner.

Hierarchical overlapping coordination (HOC) uses two or more design problem decompositions, each of them associated with different partitions of the design variables and constraints. This kind of problem decomposition may reflect, for example, matrix-type organizations structured according to product lines or physical subsystems (*object decomposition*) and the disciplines involved in the design process (*aspect decomposition*). Coordination is achieved by the exchange of information between decompositions, as explained in Section 2.1.

The mathematical formulation of HOC was first proposed in [13], and several criteria for convergence of the coordination algorithm under linear equality and inequality constraints were developed in [13] and [23]. Convergence criteria developed in those articles are computationally difficult to check and possibly incorrect (see Remark 4.5). In this article, we present computationally efficient conditions that ensure the convergence of overlapping coordination under linear equality and inequality constraints.

Several researchers have proposed coordination strategies to exploit the structure of a problem associated with its decomposition. Reviews of optimization procedures that use decomposition are presented by Wagner and Papalambros [27] and Sobieszczanski-Sobieski and Haftka [25]. Recently, Nelson and Papalambros [19] presented sequentially decomposed programming (SDP) as a globally convergent coordination scheme for hierarchic systems. Other promising coordination algorithms, including concurrent subspace optimization (CSSO) [24] and collaborative optimization (CO) [4] for nonhierarchic systems, require further study of robustness and convergence properties.

Interest in HOC here is motivated by the desire to solve decomposed problems rigorously, rather than to achieve computational speed-ups.

2. HOC under linear equality constraints

In the case of linear equality constraints only, the original optimization problem can be restated in the following form:

$$\text{Min}_{\text{x}} \; f(\mathbf{x}) \text{ subject to } A\mathbf{x} = \mathbf{c}, \tag{2.1}$$

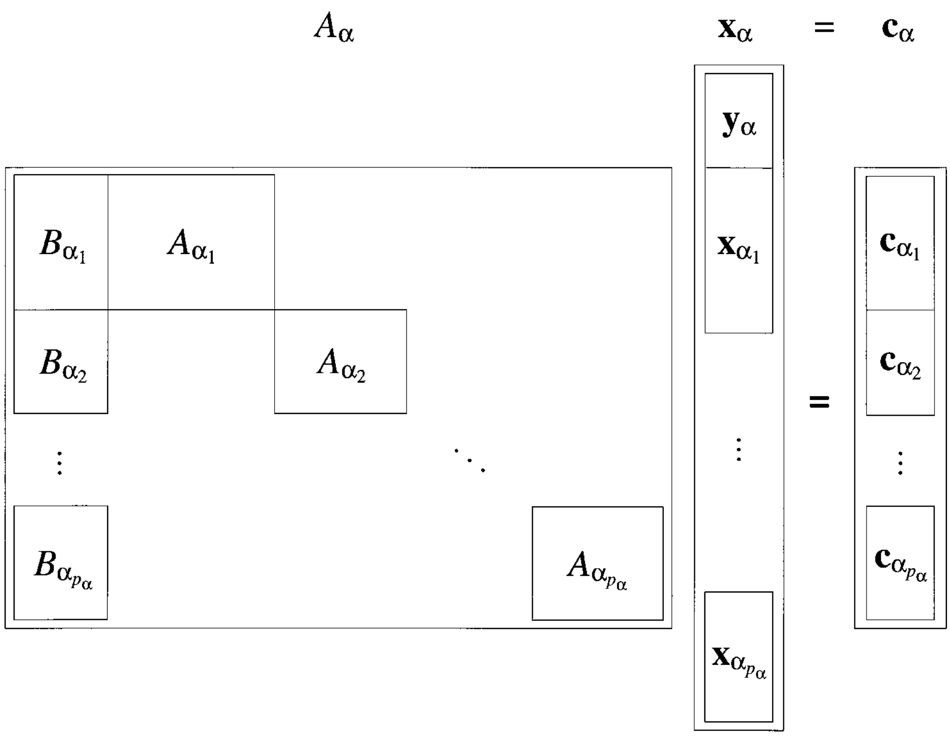

Figure 1. Block-decomposition of constraint matrix A, design vector $\mathbf{x}$, and constant vector $\mathbf{c}$.

where $f : \mathbb{R}^n \to \mathbb{R}$ is convex and differentiable, A is an $m \times n$ constraint matrix with real entries, $\mathbf{x} \in \mathbb{R}^n$ is the vector of optimization variables, and $\mathbf{c} \in \mathbb{R}^m$ is a constant vector. We assume that the above problem has a nonempty solution set.

Suppose that the columns and rows of A (and correspondingly the components of $\mathbf{x}$ and $\mathbf{c}$) can be reordered to generate a block-angular decomposition as represented by A_α in figure 1. We refer to it as the α-decomposition of the problem. In figure 1, $\mathbf{x}_\alpha$ is the vector of reordered design variables, $\mathbf{c}_\alpha$ is the reordered vector $\mathbf{c}$, and $\mathbf{y}_\alpha$ is the vector of n_α linking variables for the α-decomposition. The linking variables for the α-decomposition will be referred to as α-linking variables, and the number of subproblems in the α-decomposition (diagonal blocks in the figure) is given by p_α. More explicitly,

$$\mathbf{y}_\alpha := \begin{pmatrix} x_{\alpha(1)} \\ \vdots \\ x_{\alpha(n_\alpha)} \end{pmatrix} \quad \text{and} \quad \mathbf{c}_\alpha := \begin{pmatrix} \mathbf{c}_{\alpha 1} \\ \vdots \\ \mathbf{c}_{\alpha_{p_\alpha}} \end{pmatrix}.$$

$\mathbf{x}_{\alpha_i}$ is the vector of local variables associated with block A_{α_i}, i.e., with subproblem α_i for $i = 1, 2, \ldots, p_\alpha$. We note that the reordered matrix A_α consists of a "side" block of columns, corresponding to the α-linking variables, and diagonal blocks $A_{\alpha_1}, A_{\alpha_2}, \ldots, A_{\alpha_{p_\alpha}}$.

We assume that Problem 2.1 can be decomposed in two or more different ways (say, α-, β-, ... decompositions). Model-based decomposition methods [17, 18] can be used to produce such decompositions as described in Section 2.2. Although the results of this article can be generalized to three or more decompositions, we will consider only two problem decompositions (α and β) to simplify notations and proofs.

Typically, system design problems have to be formulated as multicriteria optimization problems. The criteria may correspond to the various problem aspects (e.g., system performance, cost, durability, weight, or dynamic response) or to design objectives for each subsystem. A monotonic value function [8] may be then used to combine dissimilar criteria to generate a design point or Pareto set. In aspect or object decomposition, this value function is separable according to aspects or subsystems, respectively. In weakly-connected model decompositions, a separable value function could be constructed to match the model decompositions as proposed in [9].

Under the assumption that the objective function f is α-additively separable,[1] Problem 2.1 takes the following form:

$$\begin{aligned} &\text{Min}_{\mathrm{x}} \ f_{\alpha_0}(\mathbf{y}_\alpha) + \sum_{i=1}^{p_\alpha} f_{\alpha_i}(\mathbf{y}_\alpha, \mathbf{x}_{\alpha_i}) \\ &\text{subject to } B_{\alpha_i}\mathbf{y}_\alpha + A_{\alpha_i}\mathbf{x}_{\alpha_i} = \mathbf{c}_{\alpha_i}, \quad i = 1, \ldots, p_\alpha. \end{aligned} \tag{2.2}$$

For a given vector $\mathbf{d}_\alpha \in \mathbb{R}^{n_\alpha}$, fixing the α-linking variables $\mathbf{y}_\alpha = \mathbf{d}_\alpha$ in (2.2) results in the following Problem α:

$$\begin{aligned} &\textbf{Problem } \alpha: \\ &\quad \text{For each } i = 1, \ldots, p_\alpha, \\ &\quad \text{Min}_{\mathrm{x}_{\alpha_i}} \ f_{\alpha_i}(\mathbf{d}_\alpha, \mathbf{x}_{\alpha_i}) \ \text{ subject to } A_{\alpha_i}\mathbf{x}_{\alpha_i} = \mathbf{c}_{\alpha_i} - B_{\alpha_i}\mathbf{d}_\alpha. \end{aligned} \tag{2.3}$$

Problem α can be solved by solving p_α independent uncoupled subproblems. Similarly, Problem β can be defined and solved for a β-decomposition after fixing the β-linking variables.

2.1. Generic HOC algorithm

The generic hierarchical overlapping coordination algorithm can be described for the case of two decompositions (α and β) as follows:

Step 1. Fix linking variables $\mathbf{y}_\alpha$, and solve Problem α by solving the p_α independent subproblems given in (2.3).
Step 2. Fix linking variables $\mathbf{y}_\beta$ to their values determined in **Step** 1, and solve Problem β by solving p_β independent subproblems.
Step 3. Go to **Step** 1 with the fixed values of α-linking variables determined in **Step** 2.
Step 4. Repeat these steps until convergence is achieved.

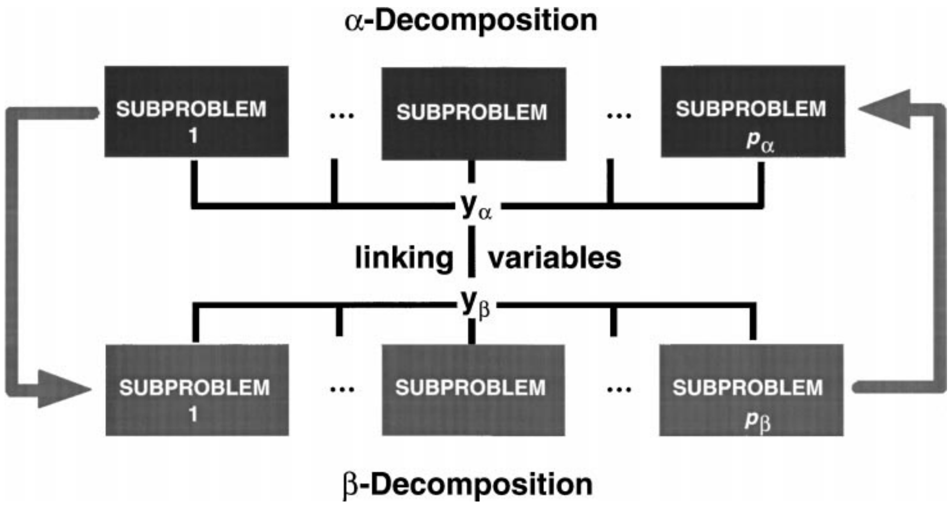

Figure 2. Flow of information in hierarchical overlapping coordination.

Thus, in the HOC algorithm, the linking variables for one of the decompositions are fixed at values that result from the solution of a number of independent subproblems associated with the previous decomposition. The flow of information between decompositions is represented in the diagram of figure 2.

Remark 2.1. The accumulation point achieved in **Step** 4 is not necessarily an optimal solution of Problem 2.1. A sufficient condition guaranteeing the convergence to a solution of the original problem will be developed in the following sections.

2.2. Finding decompositions of a design problem

Hierarchical overlapping coordination entails identifying hierarchical decompositions of the design model, i.e., groups of design submodels (or modules) that exchange information in an acyclic manner. The flow of information among modules can be then represented with a graph without circuits—a tree. Once design information is fixed at a given level, design tasks at the level below can be carried out independently. Relying on the engineer's insight to recognize a decomposition of a large multidisciplinary design model may not always be possible, so several computational techniques have been devised for hierarchical and sequential decomposition of design processes and problems.

Sequential decomposition techniques identify and arrange modules that contain design tasks that are strongly connected to minimize feedback between modules [1, 15, 21, 26]. These techniques reorder the so-called design structure matrix[2] in a block-triangular form to generate the best computational sequence. The resulting partition is nonhierarchical if feedback cannot be avoided. Feed-forward structures are still not appropriate for hierarchical decomposition and coordination because the modules may not be separable. Kroo and his collaborators [1, 11] have proposed using auxiliary variables and compatibility constraints for hierarchical decomposition of design models whose reordered structure matrix presents both feed-backward and feed-forward connections.

Similar ideas have been applied to hierarchical decomposition of design processes and problems. Kusiak proposed in [12] a branch-and-bound algorithm to partition an overall design task into subtasks with minimal interdependence, allowing concurrency of the design process. For a given design problem, a matrix called functional dependence table (FDT) can be constructed as a Boolean matrix representing the dependence of design constraint functions on variables. The (i, j)-th entry of the FDT is one if the i-th constraint depends on the j-th variable and zero otherwise.[3] A decomposition of the given design problem can be achieved by reordering rows and columns of the FDT corresponding to the constraints and variables, respectively. The decomposition algorithm proposed in Michelena and Papalambros [18] uses a hypergraph representation of the design model, which is then optimally partitioned into weakly connected subgraphs that can be identified with subproblems. An implementation of this decomposition algorithm is available on the web [16]. Design variables are represented by the hypergraph edges, whereas design constraints interrelating these variables are represented by the nodes. These constraints may be given as algebraic equations, response surfaces or look-up tables, or evaluated using simulation modules. The formulation can account for computational demands and resources as well as the strength of interdependencies between modules in the model, using weights in the graph.

The above hierarchical decomposition algorithms can be also used to identify clusters of submodels of an already partitioned design model. Note, however, that a highly coupled model might not be decomposable at all; that is, the number of linking variables would be too large in relation to the total number of variables. Optimization by decomposition, including HOC, is not appropriate in these cases. HOC is a promising method only if the sparsity of the model is such that two or more weakly connected partitions can be identified. In general, a decomposable model is characterized by having a small set of linking variables, and it is very likely to have multiple decompositions.

In practical design situations, one may not be able to rearrange the order of evaluation of design modules. A linking variable may actually be the output of a design module. Hierarchical decomposition is still possible by adding auxiliary variables and enforcing compatibility constraints within the corresponding subproblem, as suggested in [1, 2, 11] for nonhierarchical systems. This approach is equivalent to constraining a residual on the value of the output linking variable as in the individual discipline feasible (IDF) formulation in [5].

2.3. Optimality conditions

The Lagrange multiplier theorem for linear equality constraints [3, Proposition 3.4.1] states that $\mathbf{x}^* \in \mathbb{R}^n$ is a solution to Problem 2.1 if and only if there exists a vector $\boldsymbol{\lambda} \in \mathbb{R}^m$ such that

$$\nabla f^t(\mathbf{x}^*) + A^t\boldsymbol{\lambda} = \mathbf{0}. \tag{2.4}$$

In contrast to the case of nonlinear constraints, this optimality condition is valid even without the regularity assumption on $\mathbf{x}^*$. This is a consequence of Farkas' Lemma for polyhedral sets [3, page 292].

Condition 2.4 is equivalent to

$$\nabla f^t(\mathbf{x}^*) = -A^t \boldsymbol{\lambda},$$

which can be rephrased as

"$\nabla f^t(\mathbf{x}^*)$ belongs to the row space $\mathrm{RS}(A)$."

Let $\mathbf{e}_i \in \mathbb{R}^n$ be the i-th standard row vector whose i-th component is one and all other components are zero. Once an α-decomposition and a β-decomposition are given, define the indicator matrices H_α and H_β by

$$H_\alpha := \begin{pmatrix} \mathbf{e}_{\alpha(1)} \\ \mathbf{e}_{\alpha(2)} \\ \vdots \\ \mathbf{e}_{\alpha(n_\alpha)} \end{pmatrix}, \quad H_\beta := \begin{pmatrix} \mathbf{e}_{\beta(1)} \\ \mathbf{e}_{\beta(2)} \\ \vdots \\ \mathbf{e}_{\beta(n_\beta)} \end{pmatrix}. \tag{2.5}$$

These are unique $n_\alpha \times n$ and $n_\beta \times n$ matrices having ones and zeros as their entries such that

$$H_\alpha \mathbf{x} = \mathbf{y}_\alpha, \quad H_\beta \mathbf{x} = \mathbf{y}_\beta.$$

Define K_α, K_β and $K_{\alpha\beta}$ as follows:

$$K_\alpha := \begin{pmatrix} A \\ H_\alpha \end{pmatrix}, \quad K_\beta := \begin{pmatrix} A \\ H_\beta \end{pmatrix}, \quad K_{\alpha\beta} := \begin{pmatrix} A \\ H_\alpha \\ H_\beta \end{pmatrix}.$$

Problem α, with fixed values for the α-linking variables $\mathbf{y}_\alpha = \mathbf{d}_\alpha$, can be defined as

$$\mathrm{Min}_{\mathrm{x}}\ f(\mathbf{x}) \text{ subject to } A\mathbf{x} = \mathbf{c} \quad \text{and} \quad H_\alpha \mathbf{x} = \mathbf{d}_\alpha. \tag{2.6}$$

One notes that $\mathbf{x}^*_\alpha \in \mathbb{R}^n$ is a solution to Problem α if and only if there exists vectors $\boldsymbol{\lambda}_\alpha \in \mathbb{R}^m$ and $\boldsymbol{\mu}_\alpha \in \mathbb{R}^{n(\alpha)}$ such that

$$\nabla f^t(\mathbf{x}^*_\alpha) + A^t \boldsymbol{\lambda}_\alpha + H^t_\alpha \boldsymbol{\mu}_\alpha = 0. \tag{2.7}$$

This optimality condition can be rephrased as

"$\nabla f^t(\mathbf{x}^*_\alpha)$ belongs to the row space $\mathrm{RS}(K_\alpha)$."

Analogously, $\mathbf{x}^*_\beta$ is a solution to Problem β if and only if

"$\nabla f^t(\mathbf{x}^*_\beta)$ belongs to the row space $\mathrm{RS}(K_\beta)$."

2.4. *Properties of HOC*

The following properties of HOC were observed and proved in [13]. Note that they ensure convergence of the HOC algorithm.

1. If the HOC algorithm is started with a feasible point $\mathbf{x}_0$, then at each stage of the process, problem α and problem β will have nonempty feasible domains.
2. If the sequences $\{\mathbf{x}_{\alpha_i}\}_{i=1}^\infty$ and $\{\mathbf{x}_{\beta_i}\}_{i=1}^\infty$ result from solving problem α and problem β, respectively, and $f^{\min} := \min\{f(\mathbf{x}) \mid A\mathbf{x} = \mathbf{c}\}$, then
   (a) $f(\mathbf{x}_{\alpha_i}) \geq f(\mathbf{x}_{\beta_i}) \geq f(\mathbf{x}_{\alpha_{(i+1)}})$
   (b) $\lim_{i\to\infty} f(\mathbf{x}_{\alpha_i}) = \lim_{i\to\infty} f(\mathbf{x}_{\beta_i}) = f^* \geq f^{\min}$
3. Any accumulation point $\mathbf{x}^*$ of either $\{\mathbf{x}_{\alpha_i}\}_{i=1}^\infty$ or $\{\mathbf{x}_{\beta_i}\}_{i=1}^\infty$ solves both problem α and problem β.

3. Conditions for convergence under linear equality constraints

Once α- and β-decompositions of the optimization problem in (2.1) are obtained, let $\{x_{\alpha_i}\}_{i=1}^\infty$ and $\{x_{\beta_i}\}_{i=1}^\infty$ be the sequences obtained by applying the generic HOC algorithm to these decompositions as described in Section 2.1. Theorem 3.1 below gives a sufficient condition for these sequences to converge to a minimum of Problem 2.1 in terms of the row spaces $\mathrm{RS}(A)$, $\mathrm{RS}(K_\alpha)$ and $\mathrm{RS}(K_\beta)$.

Theorem 3.1. *Let* $\mathbf{x}^*$ *be an accumulation point of* $\{x_{\alpha_i}\}_{i=1}^\infty$ *or* $\{x_{\beta_i}\}_{i=1}^\infty$. *If*

$$RS(A) = RS(K_\alpha) \cap RS(K_\beta),$$

then $\mathbf{x}^*$ *is a solution to the optimization problem in* (2.1).

Proof: By Property 3 of Section 2.4, $\mathbf{x}^*$ solves both Problem α and Problem β. Therefore,

$$\nabla f^t(\mathbf{x}^*) \in \mathrm{RS}(K_\alpha) \text{ and } \nabla f^t(\mathbf{x}^*) \in \mathrm{RS}(K_\beta).$$

Since $\mathrm{RS}(A) = \mathrm{RS}(K_\alpha) \cap \mathrm{RS}(K_\beta)$, one gets $\nabla f^t(\mathbf{x}^*) \in \mathrm{RS}(A)$, which implies $\mathbf{x}^*$ is a solution to the original optimization problem. □

Although Theorem 3.1 offers a conceptually clear sufficient condition for the convergence of the HOC, it involves the algorithmic process of computing the intersection of the two vector spaces $\mathrm{RS}(K_\alpha)$ and $\mathrm{RS}(K_\beta)$. The computational cost associated with this process can be fairly high. As an attempt to obtain a computationally efficient HOC convergence condition, we prove in Theorem 3.2 that a certain matrix rank condition implies the convergence condition of Theorem 3.1.

Theorem 3.2. *Let r be the rank of A and $\hat{A}$ be an $r \times n$ submatrix of A with full row rank. If the matrix*

$$\hat{K}_{\alpha\beta} := \begin{pmatrix} \hat{A} \\ H_\alpha \\ H_\beta \end{pmatrix}$$

has full row rank, then $RS(A) = RS(K_\alpha) \cap RS(K_\beta)$.

Proof: Clearly, $\mathrm{RS}(A) \subset \mathrm{RS}(K_\alpha)$ and $\mathrm{RS}(A) \subset \mathrm{RS}(K_\beta)$. Therefore,

$$\mathrm{RS}(A) \subset \mathrm{RS}(K_\alpha) \cap \mathrm{RS}(K_\beta).$$

To show the reverse inclusion, choose an arbitrary $\mathbf{v} \in \mathrm{RS}(K_\alpha) \cap \mathrm{RS}(K_\beta)$. Let $\mathbf{v}_1, \ldots, \mathbf{v}_r$ be the row vectors of $\hat{A}$, and $\mathbf{e}_i \in \mathbb{R}^n$ be the i-th standard row vector. The full row rank condition on

$$\hat{K}_{\alpha\beta} = \left(\mathbf{v}_1, \ldots, \mathbf{v}_r, \mathbf{e}_{\alpha(1)}, \ldots, \mathbf{e}_{\alpha(n_\alpha)}, \mathbf{e}_{\beta(1)}, \ldots, \mathbf{e}_{\beta(n_\beta)}\right)^t$$

implies that $\{\mathbf{v}_1, \ldots, \mathbf{v}_r, \mathbf{e}_{\alpha(1)}, \ldots, \mathbf{e}_{\alpha(n_\alpha)}\}$ and $\{\mathbf{v}_1, \ldots, \mathbf{v}_r, \mathbf{e}_{\beta(1)}, \ldots \mathbf{e}_{\beta(n_\beta)}\}$ are bases for $\mathrm{RS}(K_\alpha)$ and $\mathrm{RS}(K_\beta)$, respectively.

$$\mathbf{v} \in \mathrm{RS}(K_\alpha) \Longrightarrow \mathbf{v} = \sum_{i=1}^{r} a_i \mathbf{v}_i + \sum_{i=1}^{n_\alpha} s_i \mathbf{e}_{\alpha(i)} \text{ for some } a_i\text{'s and } s_i\text{'s in } \mathbb{R}.$$
$$\mathbf{v} \in \mathrm{RS}(K_\beta) \Longrightarrow \mathbf{v} = \sum_{i=1}^{r} b_i \mathbf{v}_i + \sum_{i=1}^{n_\beta} t_i \mathbf{e}_{\beta(i)} \text{ for some } b_i\text{'s and } t_i\text{'s in } \mathbb{R}.$$

Therefore,

$$\sum_{i=1}^{r} (a_i - b_i)\mathbf{v}_i + \sum_{i=1}^{n_\alpha} s_i \mathbf{e}_{\alpha(i)} - \sum_{i=1}^{n_\beta} t_i \mathbf{e}_{\beta(i)} = \mathbf{0}.$$

Since $\mathbf{v}_1, \ldots, \mathbf{v}_r, \mathbf{e}_{\alpha(1)}, \ldots, \mathbf{e}_{\alpha(n_\alpha)}, \mathbf{e}_{\beta(1)}, \ldots, \mathbf{e}_{\beta(n_\beta)}$ are linearly independent, one concludes

$$a_i = b_i, s_j = 0, t_k = 0, \text{ for all } i = 1, \ldots, r, j = 1, \ldots, n_\alpha, k = 1, \ldots, n_\beta.$$

This means $\mathbf{v} = \sum_{i=1}^{r} a_i \mathbf{v}_i \in \mathrm{RS}(A)$. □

Corollary 3.3. *Same notations as in Theorems* 3.1 *and* 3.2. *If the matrix $\hat{K}_{\alpha\beta}$ has full row rank, then $\mathbf{x}^*$ is a solution to the optimization problem in* (2.1).

Remark 3.4. The convergence condition given in Corollary 3.3 offers a very efficient criterion for the convergence of hierarchical overlapping coordination in terms of rank $(\hat{K}_{\alpha\beta})$. A model partitioning algorithm as in [18] should just check if rank $(\hat{K}_{\alpha\beta})$ is equal to $r + n_\alpha + n_\beta$.

The following theorem offers several interpretations of the HOC convergence condition given in Corollary 3.3.

Theorem 3.4. *With the same notations as in Theorem* 3.2, *the following three conditions are equivalent.*

1. *The matrix $\hat{K}_{\alpha\beta}$ has full row rank.*
2. *There exists no* (*nontrivial*) *linear relation exclusively among the α- and β-linking variables.*
3. *The set of α-linking variables and the set of β-linking variables are disjoint*, *and the matrix $\hat{A}_{\alpha\beta}$ obtained from $\hat{A}$ by deleting columns corresponding to the α-and β-linking variables has full row rank.*

Proof: (1) $\Longleftrightarrow$ (2): From $A\mathbf{x} = \mathbf{c}$, one can find the unique vector $\hat{\mathbf{c}} = (\hat{c}_1, \ldots, \hat{c}_r)^t$ such that $\hat{A}\mathbf{x} = \hat{\mathbf{c}}$. Let $\mathbf{v}_1, \ldots, \mathbf{v}_r$ be the row vectors of $\hat{A}$ and $\mathbf{e}_i \in \mathbb{R}^n$ be the i-th standard row vector. Note that

$$\mathbf{v}_i\mathbf{x} = \hat{c}_i, \quad \mathbf{e}_i\mathbf{x} = x_i.$$

$\hat{K}_{\alpha\beta}$ is rank-deficient. $\Longleftrightarrow$ There exists a nontrivial linear relation among the row vectors of $\hat{K}_{\alpha\beta}$.

$\Longleftrightarrow$ There exists a nontrivial linear relation

$$\sum_{i=1}^{r} a_i\mathbf{v}_i + \sum_{i=1}^{n_\alpha} s_i\mathbf{e}_{\alpha(i)} + \sum_{i=1}^{n_\beta} t_i\mathbf{e}_{\beta(i)} = 0.$$

$\Longleftrightarrow$ There exists a nontrivial linear relation

$$\sum_{i=1}^{r} a_i\mathbf{v}_i\mathbf{x} + \sum_{i=1}^{n_\alpha} s_i\mathbf{e}_{\alpha(i)}\mathbf{x} + \sum_{i=1}^{n_\beta} t_i\mathbf{e}_{\beta(i)}\mathbf{x} = 0.$$

$\Longleftrightarrow$ There exists a nontrivial linear relation

$$\sum_{i=1}^{r} a_i\hat{c}_i + \sum_{i=1}^{n_\alpha} s_i x_{\alpha(i)} + \sum_{i=1}^{n_\beta} t_i x_{\beta(i)} = 0.$$

$\Longleftrightarrow$ There exists a nontrivial linear relation exclusively among α-and β-linking variables.

(1) $\Longrightarrow$ (3): Since $\binom{H_\alpha}{H_\beta}$ has full row rank, the sets $\{\alpha(1), \ldots, \alpha(n_\alpha)\}$ and $\{\beta(1), \ldots, \beta(n_\alpha)\}$ are disjoint. By performing appropriate elementary row operations on $\hat{K}_{\alpha\beta}$, one easily concludes that $\hat{A}_{\alpha\beta}$ has full row rank.

(3) $\Longrightarrow$ (1): Straightforward. □

Theorem 3.6 below shows that, under a certain additional condition, the two conditions of Theorem 3.2 are actually equivalent.

Theorem 3.6. *Suppose that* $\hat{K}_\alpha := \binom{\hat{A}}{H_\alpha}$ *and* $\hat{K}_\beta := \binom{\hat{A}}{H_\beta}$ *have full row rank. Then,* $\mathrm{RS}(A) = \mathrm{RS}(K_\alpha) \cap \mathrm{RS}(K_\beta)$ *if and only if the matrix* $\hat{K}_{\alpha\beta} := \begin{pmatrix} \hat{A} \\ H_\alpha \\ H_\beta \end{pmatrix}$ *has full row rank.*

Proof:

($\Longleftarrow$): Shown in Theorem 3.2.
($\Longrightarrow$): Suppose $\hat{K}_{\alpha\beta}$ is rank-deficient. Let $\mathbf{v}_1, \ldots, \mathbf{v}_r$ be the row vectors of $\hat{A}$, and $\mathbf{e}_i \in \mathbb{R}^n$ be the i-th standard row vector. Since

$$\hat{K}_{\alpha\beta} = \left(\mathbf{v}_1, \ldots, \mathbf{v}_r, \mathbf{e}_{\alpha(1)}, \ldots, \mathbf{e}_{\alpha(n_\alpha)}, \mathbf{e}_{\beta(1)}, \ldots, \mathbf{e}_{\beta(n_\beta)}\right)^t$$

is rank-deficient, there exists a nontrivial linear relation among its row vectors:

$$\sum_{i=1}^{r} a_i \mathbf{v}_i + \sum_{i=1}^{n_\alpha} s_i \mathbf{e}_{\alpha(i)} + \sum_{i=1}^{n_\beta} t_i \mathbf{e}_{\beta(i)} = \mathbf{0}, \tag{3.1}$$

where the coefficients are not identically zero. In this expression, the first two sums belong to $\mathrm{RS}(K_\alpha)$ while the third sum belongs to $\mathrm{RS}(K_\beta)$. Therefore,

$$\sum_{i=1}^{r} a_i \mathbf{v}_i + \sum_{i=1}^{n_\alpha} s_i \mathbf{e}_{\alpha(i)} = -\sum_{i=1}^{n_\beta} t_i \mathbf{e}_{\beta(i)} \in \mathrm{RS}(K_\alpha) \cap \mathrm{RS}(K_\beta) = \mathrm{RS}(A).$$

Since $\mathbf{v}_i$'s and $\mathbf{e}_{\alpha(j)}$'s form a basis for $\mathrm{RS}(K_\alpha) = \mathrm{RS}(\hat{K}_\alpha)$, an arbitrary element of $\mathrm{RS}(K_\alpha)$ has a unique expression as a linear combination of these basis vectors. In particular, an element of $\mathrm{RS}(A) \subset \mathrm{RS}(K_\alpha)$ is expressed only in terms of $\mathbf{v}_i$'s. Therefore, from $\sum_{i=1}^{r} a_i \mathbf{v}_i + \sum_{i=1}^{n_\alpha} s_i \mathbf{e}_{\alpha(i)} \in \mathrm{RS}(A)$, we deduce that

$$s_1 = \cdots = s_{n_\alpha} = 0.$$

Thus, (3.1) becomes

$$\sum_{i=1}^{r} a_i \mathbf{v}_i + \sum_{i=1}^{n_\beta} t_i \mathbf{e}_{\beta(i)} = \mathbf{0}, \tag{3.2}$$

where the coefficients are not identically zero. This contradicts the full row rank condition on $\hat{K}_\beta$, and thus, $\hat{K}_{\alpha\beta}$ has to be a full row rank matrix. □

Since the row space $\mathrm{RS}(B)$ of a matrix B can be viewed as the orthogonal complement of its null space $\mathrm{NS}(B)$, the HOC convergence condition of Theorem 3.1 given in terms of the row spaces of A, K_α and K_β can be rephrased in terms of their null spaces.

For any subspace W of an inner product space V, we denote the orthogonal complement of W by $W^{\perp}$. We need the following lemma.

Lemma 3.7. *Let W_1 and W_2 be subspaces of a finite dimensional inner product space. Then*

$$(W_1 + W_2)^{\perp} = W_1^{\perp} \cap W_2^{\perp}$$
$$(W_1 \cap W_2)^{\perp} = W_1^{\perp} + W_2^{\perp}$$

Proof: Exercise 11 in [6, page 313]. □

Theorem 3.8. *Let A, K_α and K_β be the matrices defined by two decompositions of the optimization problem in* (2.1). *Then*

$$\mathrm{RS}(A) = \mathrm{RS}(K_\alpha) \cap \mathrm{RS}(K_\beta)$$

if and only if

$$\mathrm{NS}(A) = \mathrm{NS}(K_\alpha) + \mathrm{NS}(K_\beta).$$

Proof: For an arbitrary $s \times t$ matrix B, identify $\mathrm{RS}(B)$ and $\mathrm{NS}(B)$ as subspaces of $\mathbb{R}^t$, and identify a t-dimensional row vector with a t-dimensional column vector. Just note that

$$\mathrm{RS}(B)^{\perp} = \mathrm{NS}(B),$$

and apply Lemma 3.7. □

In [13] and [23], the null space condition $\mathrm{NS}(A) = \mathrm{NS}(K_\alpha) + \mathrm{NS}(K_\beta)$ was developed as a sufficient condition for convergence of HOC. A computational procedure to check the convergence of HOC based on this condition will have to compute the sum of two vector spaces, $\mathrm{N}(K_\alpha)$ and $\mathrm{N}(K_\beta)$, which is an expensive computational process. Also, this condition is sometimes difficult to work with. For instance, the following apparently incorrect statement appears in [23]:

> [23, Property 4] If the decision variables corresponding to the interaction (i.e., linking) variables $\mathbf{y}_\alpha$ and $\mathbf{y}_\beta$ are bounded by common equations, then
>
> $$\mathrm{NS}(A) \neq \mathrm{NS}(K_\alpha) + \mathrm{NS}(K_\beta).$$

Example 2 in [23] was constructed specifically to demonstrate the above Property 4. The constraints in this example are

$$\begin{aligned} x_1 + 3x_2 + 2x_3 &= c_1 \\ x_2 + 2x_3 + x_4 &= c_2 \\ x_4 + x_5 + x_6 + 2x_7 &= c_3 \\ x_5 + 3x_6 + x_7 &= c_4 \\ x_6 + x_8 + x_9 + x_{10} &= c_5 \\ x_9 + x_{10} &= c_6, \end{aligned}$$

and the linking variables are $y_\alpha = x_6$ and $y_\beta = x_4$. Based on the observation that the third constraint equation contains both y_α and y_β, the article [23] claims that this example satisfies the hypothesis of Property 4 and therefore $\mathrm{NS}(A) \neq \mathrm{NS}(K_\alpha) + \mathrm{NS}(K_\beta)$. It also presents a computation that results in the erroneous conclusion that $\mathrm{NS}(A)$ and $\mathrm{NS}(K_\alpha) + \mathrm{NS}(K_\beta)$ are different. An explicit computation using *Maple* [20] actually shows that $\mathrm{NS}(A) = \mathrm{NS}(K_\alpha) + \mathrm{NS}(K_\beta)$, and that indeed this example disproves [23, Property 4].

The above example demonstrates how difficult it can be to check the convergence criterion $\mathrm{NS}(A) = \mathrm{NS}(K_\alpha) + \mathrm{NS}(K_\beta)$ in actual computation.

4. HOC under mixed linear constraints

In this section, we extend the results of the preceding sections to the general case of HOC under mixed linear equality and inequality constraints:

$$\mathrm{Min}_x\, f(\mathbf{x}) \text{ subject to } A^I\mathbf{x} \le \mathbf{c}^I \;\text{ and }\; A^E\mathbf{x} = \mathbf{c}^E \tag{4.1}$$

where $f : \mathbb{R}^n \to \mathbb{R}$ is convex and differentiable, A^I (A^E, resp.) is an $m_I \times n$ ($m_E \times n$, resp.) constraint matrix with real entries, $\mathbf{x} \in \mathbb{R}^n$ is the vector of optimization variables, and $\mathbf{c}^I \in \mathbb{R}^{m_I}$ ($\mathbf{c}^E \in \mathbb{R}^{m_E}$, resp.) is a constant vector. Let A be the matrix $\binom{A^I}{A^E}$. The problem is assumed to have a nonempty solution set.

The HOC algorithm described in Section 2.1 applied to Problem 4.1 results in two sequences $\{\mathbf{x}_{\alpha_i}\}_{i=1}^\infty$ and $\{\mathbf{x}_{\beta_i}\}_{i=1}^\infty$. For an accumulation point $\mathbf{x}^*$ of $\{\mathbf{x}_{\alpha_i}\}_{i=1}^\infty$ or $\{\mathbf{x}_{\beta_i}\}_{i=1}^\infty$, define J_a to be the set of the indices corresponding to the active inequality constraints, i.e.,

$$J_a := \left\{ i \;\middle|\; \left(a^I_{i1}, \ldots, a^I_{in}\right)\mathbf{x}^* = c^I_i \right\},$$

where a^I_{ij} denotes the (i, j)-entry of the matrix A^I. Let $\bar{A}^I$ be the submatrix of A^I consisting of the active inequality constraints.

Define the cone $\mathrm{C}(A)$ by

$$\mathrm{C}(A) := \left\{ \mathbf{x} \mid \mathbf{x} = \sum_{i \in J_a} a_i \mathbf{v}^I_i + \sum_{i=1}^{m_E} b_i \mathbf{v}^E_i,\, a_i \ge 0 \right\}, \tag{4.2}$$

where $\mathbf{v}_i^I$ ($\mathbf{v}_i^E$, resp.) denotes the i-th row vector of A^I (A^E, resp.). Also, define the induced cones $\mathrm{C}(K_\alpha)$ and $\mathrm{C}(K_\beta)$ as follows:

$$\mathrm{C}(K_\alpha) := \left\{ \mathbf{x} \mid \mathbf{x} = \sum_{i \in J_a} a_i \mathbf{v}_i^I + \sum_{i=1}^{m_E} b_i \mathbf{v}_i^E + \sum_{i=1}^{n_\alpha} s_i \mathbf{e}_{\alpha(i)},\, a_i \geq 0 \right\},$$
$$\mathrm{C}(K_\beta) := \left\{ \mathbf{x} \mid \mathbf{x} = \sum_{i \in J_a} a_i \mathbf{v}_i^I + \sum_{i=1}^{m_E} b_i \mathbf{v}_i^E + \sum_{i=1}^{n_\beta} t_i \mathbf{e}_{\beta(i)},\, a_i \geq 0 \right\}.$$

The Lagrange multiplier theorem for linear constraints [3, Proposition 3.4.1] states that $\mathbf{x}^* \in \mathbb{R}^n$ is a solution to Problem 4.1 if and only if there exists a **nonnegative** vector $\boldsymbol{\lambda}^I \geq \mathbf{0}$ and a vector $\boldsymbol{\lambda}^E$ such that

$$\nabla f^t(\mathbf{x}^*) + \bar{A}^{I^t} \boldsymbol{\lambda}^I + A^{E^t} \boldsymbol{\lambda}^E = \mathbf{0}. \tag{4.3}$$

As in the case of equality constraints, this result is valid even when $\mathbf{x}^*$ is not regular [3, page 292].

Condition 4.3 is equivalent to

$$-\nabla f^t(\mathbf{x}^*) = \bar{A}^{I^t} \boldsymbol{\lambda}^I + A^{E^t} \boldsymbol{\lambda}^E, \quad \boldsymbol{\lambda}^I \geq 0, \tag{4.4}$$

which can be rephrased as

"$-\nabla f^t(\mathbf{x}^*)$ belongs to the cone $\mathrm{C}(A)$."

For fixed values of the α-linking variables $\mathbf{y}_\alpha = \mathbf{d}_\alpha$, Problem α can be defined as

$$\mathrm{Min}_x\, f(\mathbf{x}) \text{ subject to } A^I \mathbf{x} \leq \mathbf{c}^I, \quad A^E \mathbf{x} = \mathbf{c}^E \text{ and } H_\alpha \mathbf{x} = \mathbf{d}_\alpha. \tag{4.5}$$

Using the above reasoning, one sees that $\mathbf{x}_\alpha^*$ is a solution to Problem α if and only if

"$-\nabla f^t(\mathbf{x}_\alpha^*)$ belongs to the cone $\mathrm{C}(K_\alpha)$."

Analogously, $\mathbf{x}_\beta^*$ is a solution to Problem β if and only if

"$-\nabla f^t(\mathbf{x}_\beta^*)$ belongs to the cone $\mathrm{C}(K_\beta)$."

The following theorem offers an analogue of Theorem 3.1.

Theorem 4.1. *Let $\mathbf{x}^*$ be an accumulation point of $\{x_{\alpha_i}\}_{i=1}^\infty$ or $\{x_{\beta_i}\}_{i=1}^\infty$. If $\mathrm{C}(A) = \mathrm{C}(K_\alpha) \cap \mathrm{C}(K_\beta)$, then $\mathbf{x}^*$ is a solution to the optimization problem in* (4.1).

Proof: By Property 3 of Section 2.4, $\mathbf{x}^*$ solves both Problem α and Problem β. Therefore,

$$-\nabla f^t(\mathbf{x}^*) \in \mathrm{C}(K_\alpha) \text{ and } -\nabla f^t(\mathbf{x}^*) \in \mathrm{C}(K_\beta).$$

Since $\mathrm{C}(A) = \mathrm{C}(K_\alpha) \cap \mathrm{C}(K_\beta)$, one gets $-\nabla f^t(\mathbf{x}^*) \in \mathrm{C}(A)$, which implies $\mathbf{x}^*$ is a solution to the original optimization problem. □

The HOC convergence condition stated in Theorem 4.1 cannot be practically used because one has to know a priori the accumulation point $\mathbf{x}^*$ and the set J_a of active constraints in order to compute the cones $\mathrm{C}(A)$, $\mathrm{C}(K_\alpha)$ and $\mathrm{C}(K_\beta)$.

As an analogue of Theorem 3.2, Theorem 4.2 below fixes this problem and provides a new sufficient condition for the convergence of HOC. This condition does not rely on the accumulation point $\mathbf{x}^*$.

Theorem 4.2. *Let r be the rank of A and $\hat{A}$ be an $r \times n$ submatrix of A with full row rank. If the matrix*

$$\hat{K}_{\alpha\beta} := \begin{pmatrix} \hat{A} \\ H_\alpha \\ H_\beta \end{pmatrix}$$

has full row rank, then $\mathrm{C}(A) = \mathrm{C}(K_\alpha) \cap \mathrm{C}(K_\beta)$.

Proof: Clearly, $\mathrm{C}(A) \subset \mathrm{C}(K_\alpha)$ and $\mathrm{C}(A) \subset \mathrm{C}(K_\beta)$. Therefore, $\mathrm{C}(A) \subset \mathrm{C}(K_\alpha) \cap \mathrm{C}(K_\beta)$.

To show the reverse inclusion, choose an arbitrary $\mathbf{v} \in \mathrm{C}(K_\alpha) \cap \mathrm{C}(K_\beta)$. Let $\mathbf{v}_1, \ldots, \mathbf{v}_r$ be the row vectors of $\hat{A}$, and $\mathbf{e}_i \in \mathbb{R}^n$ be the i-th standard row vector. Since

$$\mathbf{v} \in \mathrm{C}(K_\alpha) \Longrightarrow \mathbf{v} = \sum_{i \in J_a} a_i \mathbf{v}_i^I + \sum_{i=1}^{m_E} b_i \mathbf{v}_i^E + \sum_{i=1}^{n_\alpha} s_i \mathbf{e}_{\alpha(i)},\ a_i \geq 0,$$

$$\mathbf{v} \in \mathrm{C}(K_\beta) \Longrightarrow \mathbf{v} = \sum_{i \in J_a} d_i \mathbf{v}_i^I + \sum_{i=1}^{m_E} e_i \mathbf{v}_i^E + \sum_{i=1}^{n_\beta} t_i \mathbf{e}_{\beta(i)},\ d_i \geq 0,$$

we have

$$\sum_{i \in J_a} (a_i - d_i)\mathbf{v}_i^I + \sum_{i=1}^{m_E} (b_i - e_i)\mathbf{v}_i^E + \sum_{i=1}^{n_\alpha} s_i \mathbf{e}_{\alpha(i)} - \sum_{i=1}^{n_\beta} t_i \mathbf{e}_{\beta(i)} = 0. \tag{4.6}$$

Since $\mathbf{v}_1, \ldots, \mathbf{v}_r$ from a basis for the row space of $\left(\begin{smallmatrix} A^1 \\ A^E \end{smallmatrix}\right)$, there exist $\gamma_1, \ldots, \gamma_r \in \mathbb{R}$ such that

$$\sum_{i \in J_a} (a_i - d_i)\mathbf{v}_i^I + \sum_{i=1}^{m_E} (b_i - e_i)\mathbf{v}_i^E = \sum_{i=1}^{r} \gamma_i \mathbf{v}_i.$$

Therefore, (4.6) becomes

$$\sum_{i=1}^{r} \gamma_i \mathbf{v}_i + \sum_{i=1}^{n_\alpha} s_i \mathbf{e}_{\alpha(i)} - \sum_{i=1}^{n_\beta} t_i \mathbf{e}_{\beta(i)} = 0.$$

Since

$$\hat{K}_{\alpha\beta} = \left(\mathbf{v}_1, \ldots, \mathbf{v}_r, \mathbf{e}_{\alpha(1)}, \ldots, \mathbf{e}_{\alpha(n_\alpha)}, \mathbf{e}_{\beta(1)}, \ldots, \mathbf{e}_{\beta(n_\beta)}\right)^t$$

has full row rank, $\mathbf{v}_1, \ldots, \mathbf{v}_r, \mathbf{e}_{\alpha(1)}, \ldots, \mathbf{e}_{\alpha(n_\alpha)}, \mathbf{e}_{\beta(1)}, \ldots, \mathbf{e}_{\beta(n_\beta)}$ are linearly independent. Therefore,

$$\gamma_i = 0, s_j = 0, t_k = 0, \text{ for all } i = 1, \ldots, r, \; j = 1, \ldots, n_\alpha, k = 1, \ldots, n_\beta,$$

and thus

$$\mathbf{v} = \sum_{i \in J_a} a_i \mathbf{v}_i^I + \sum_{i=1}^{m_E} b_i \mathbf{v}_i^E, a_i \geq 0.$$

This implies that $\mathbf{v} \in \mathrm{C}(A)$ □

Theorem 4.2 combined with Theorem 4.1 immediately implies the following Corollary.

Corollary 4.3. *Same notations as in Theorems* 4.1 *and* 4.2. *If* $\hat{K}_{\alpha\beta}$ *has full row rank*, *then* $\mathbf{x}^*$ *is a solution to the optimization problem in* (4.1).

Remark 4.4. $\hat{K}_{\alpha\beta}$ has full row rank only if the sets of α- and β- linking variables are disjoint and only if the sum of the rank of A plus the total number of linking variables is less than or equal to the total number of variables.

Remark 4.5. In [23, Property 13], an HOC convergence condition under inequality constraints was given in terms of null spaces, which does not rely on the accumulation point $\mathbf{x}^*$, either. Suppose that α- and β-decompositions are given for the problem

$$\text{Min}_x \; f(\mathbf{x}) \text{ subject to } A\mathbf{x} \leq \mathbf{c}, \tag{4.7}$$

in which A is an $m \times n$ matrix. Define matrices $\tilde{A}$, $\tilde{K}_\alpha$ and $\tilde{K}_\beta$ by

$$\tilde{A} := (A \; I_m), \tilde{K}_\alpha := \begin{pmatrix} A & I_m \\ H_\alpha & \mathbf{0} \end{pmatrix}, \quad \tilde{K}_\beta := \begin{pmatrix} A & I_m \\ H_\beta & \mathbf{0} \end{pmatrix}.$$

It was claimed in [23] that the condition

$$\mathrm{NS}(\tilde{A}) = \mathrm{NS}(\tilde{K}_\alpha) + \mathrm{NS}(\tilde{K}_\beta)$$

guarantees the appropriate convergence of HOC for the problem in (4.7).

However, this assertion does not seem to be correct. First, note that the three matrices $\tilde{A}$, $\tilde{K}_\alpha$ and $\tilde{K}_\beta$ have full row rank, and due to Theorem 3.6, the above null space condition is equivalent to the full row rank condition on the matrix

$$\begin{pmatrix} A & I_m \\ H_\alpha & \mathbf{0} \\ H_\beta & \mathbf{0} \end{pmatrix}.$$

This matrix has full row rank if and only if $\binom{H_\alpha}{H_\beta}$ has full row rank, which is true if and only if the sets of α- and β-linking variables are disjoint. However, the disjointness of α- and β-linking variables is not enough to guarantee the convergence of HOC.

5. Computational results

5.1. Obtaining two distinct decompositions

To solve an optimization problem P by the HOC algorithm, two distinct (α-, β-) decompositions of P satisfying the sufficient condition for convergence of HOC (Corollary 4.3) can be found by the following heuristic:

1. Obtain an α-decomposition by applying the hypergraph-based model decomposition algorithm (developed in [18] and described in Section 2.2) to problem P.
2. Obtain a β-decomposition by penalizing[4] the α-linking variables so that the disjointness of the set of α-linking variables and the set of β-linking variables is accomplished, as required by the convergence condition (part (3) of Theorem 3.5). If the two sets of linking variables are not disjoint, then go back to Step 1 and obtain a new α-decomposition after penalizing the common linking variables.
3. Check if the resulting α-and β-decompositions satisfy the convergence condition in Corollary 4.3. If the convergence condition is not satisfied, then go back to Step 1 and obtain a new α-decomposition after penalizing one of the interdependent linking variables (see part (2) of Theorem 3.5).

The HOC convergence condition in Step 3 above consists of checking the full-rankness of the matrix $\hat{K}_{\alpha\beta}$. This is typically achieved by applying Gaussian Elimination to $\hat{K}_{\alpha\beta}$ and finding its row echelon form. Gaussian Elimination is well studied in numerical linear algebra, and many efficient algorithms have been developed. In general, its complexity is cubic in terms of the size of the matrix, although some inherent structure of the matrix frequently lowers this complexity. Many computer algebra packages include efficient implementations of these algorithms (e.g., *rank* command in *Maple*).

5.2. Illustrative examples

We consider a family of quadratic programming (QP) problems of various sizes. The smallest QP problem P_1 has 25 variables and 21 linear constraints (19 equalities and 2 inequalities) with a strictly convex, additively separable objective function. Thus, the FDT of problem

```
          DESIGN VARIABLES

          1000000000111111111222222
          3123456789012456789012345

     1    .%%%%%...................
     2    ...%..%..................
D    3    .%.....%%................
E    4    ...%.....%%..............
S    5    .......%...%.............
I    6    ..........%.%............
G    7    *%%%.....................
N    8    *%.........%.%...........
     9    .......%......%..........
    10    ..........%....%.........
R   11    .%.....%......%.%........
E   12    ...%......%....%.%.......
L   20    .....%..%%...............
A   21    ...%........%............
T   13    *.................@......
I   14    ..................@@.....
O   15    *...................@....
N   16    ...................@@@...
S   17    ..................@...@..
    18    ..................@....@.
    19    ..................@.....@

                   (a)
```

```
          DESIGN VARIABLES

          0001111000000111111222222
          3960268124578134579012345

     2    *.%......................
     4    **.%.....................
     6    ...%%....................
    10    ...%.%...................
    12    *..%.%%..................
    21    *...%....................
     1    *......@@@@..............
     3    .......@...@@............
     5    ...........@.@...........
     7    *......@@.....@..........
     8    .......@.....@@@.........
     9    ...........@....@........
    11    .......@...@....@@.......
    13    ..............@...@......
    14    ..................@@.....
    15    ..............@.....@....
    16    ...................@@@...
    17    ..................@...@..
    18    ..................@....@.
    19    ..................@.....@
    20    .*........@.@............

                   (b)
```

Figure 3. Decompositions of example problem P_1: (a) α-decomposition and (b) β-decomposition.

P_1 is a 21×25 table. P_1 actually is a QP approximation of a design problem of a hybrid electric powertrain that uses a battery pack and a flywheel as energy sources [10]. The largest QP problem P_9 has 500 variables and 420 linear constraints (380 equalities and 40 inequalities).

Figure 3 shows the reordered FDTs for the α-and β-decompositions obtained by applying the above decomposition process to the problem P_1. *Maple* [20] was used to verify that these two decompositions do satisfy the convergence condition in Corollary 4.3. The α-decomposition in figure 3 has two subproblems and one linking variable (x_{13}), whereas the β-decomposition has two subproblems and two linking variables (x_3 and x_9).

Once the α- and β-decompositions are determined, the QP subproblems have to be solved repeatedly. *QPOPT*, the QP solver by Gill et al. [7], was used. The *MATLAB* [14] program implementing the HOC algorithm calls a *QPOPT* MEX function to solve the QP subproblems. The HOC iteration process stops if the relative difference between the values of the objective function for two consecutive iterations is less than a preset tolerance value. The tolerance value used for the computation was 10^{-5}.

To compare the effectiveness of the HOC algorithm with the ordinary All-At-Once (AAO) approach (i.e., one not using decompositions), the problems were solved in both ways. Even though the original problem yields a sparse matrix, each of the subproblems in the HOC

Table 1. CPU-runtimes for QP problems of various sizes.

				All at once		Hierarchical overlapping coordination			
Prob	No. var	No. constr	No. subpr	Objective	runtime[a]	Serial-Objectives	Parallel-runtime[a]	No. runtime[a]	iterations
P_1	25	21	2	165.52777	0.026	165.52777	0.123	0.096	4
P_2	50	42	4	331.05554	0.156	331.05555	0.246	0.100	4
P_3	75	63	6	496.58329	0.453	496.58333	0.340	0.110	4
P_4	100	84	8	662.11103	0.980	662.11111	0.443	0.110	4
P_5	125	105	10	827.63875	1.810	827.63889	0.560	0.103	4
P_6	200	168	16	1324.2218	7.036	1324.2222	0.896	0.123	4
P_7	250	210	20	1655.2772	13.66	1655.2777	1.163	0.126	4
P_8	375	315	30	2482.9157	44.12	2482.9166	1.716	0.113	4
P_9	500	420	40	3310.5590	103.2	3310.5555	2.260	0.123	4

[a]Runtime is measured in CPU seconds on a Sun UltraSpace 1.

may not be really sparse, so a sparse optimizer was not used with either approach. An AAO approach with sparse optimizers may turn out to be comparable in performance with HOC. However, HOC may increase computational efficiency of any general-purpose optimizer in the case of sparse problems.

The results for P_1 and for the other QP problems of larger sizes are shown in Table 1. Runtime was measured in CPU seconds on a stand-alone Sun UltraSparc 1. Runtimes only include *QPOPT* function calls, excluding I/O and data transfer between α- and β-decompositions.

The algorithm terminates after four iterations in all nine cases. Each runtime represents the average of runtimes measured for five separate runs of the algorithm; the times of the five runs were consistently close. Serial-runtime is measured for the HOC computation with the subproblems solved sequentially, whereas parallel-runtime is measured for the HOC computation with the subproblems simulated as if they were solved in parallel.

HOC has lower parallel-runtimes than the ordinary AAO algorithm except for problem P_1. HOC has lower serial-runtimes than the AAO algorithm except for problems P_1 and P_2. Each of the two decompositions for P_1 has two subproblems. Each of the two decompositions for P_9 has 40 subproblems. Note that HOC for P_9 has parallel- and serial-runtimes that are 840 and 45 times faster than the AAO runtime, respectively. This is a promising result in that HOC may not only allow rigorous solution of decomposed problems but also provide computational benefits.

6. Conclusion

Hierarchical overlapping coordination can be used if the problem can be partitioned in a set of loosely connected subproblems. The approach is attractive as a solution method for large multidisciplinary design problems. In this article we showed that convergence conditions may be tested in a relatively simple and inexpensive manner. The illustrative example gives also some encouragement that computational solution costs may be substantially reduced. Future research is required to validate whether the computational advantage can be enjoyed for a larger variety of problems.

Acknowledgment

This research was supported by the U.S. Army Automotive Research Center for Modeling and Simulation of Ground Vehicle under contract No. DAAE07-94-C-R094, and by the Oakland University Foundation. This support is gratefully acknowledged. The authors wish to thank Mr. Hyung Min Kim for his assistance with the *MATLAB* implementation of the HOC algorithm.

Notes

1. In general, HOC can be used if the objective function can be written as a monotonic function of local objective functions derived from the α- and β-decompositions.
2. A design structure matrix is used to represent precedence relations between design tasks. A nonzero (i, j)-th entry in a design structure matrix indicates that task j contributes information to task i.
3. In the context of design processes, an FDT is referred as the task-parameter incidence matrix. A (i, j)-th entry in a design incidence matrix indicates that information j is needed to perform task i.
4. A variable is penalized when it is not desirable to have the variable as a linking variable. This can be achieved by assigning a high weight to the corresponding hyperedge in the hypergraph-based model decomposition algorithm described in [18].

References

1. A. Altus, I. Kroo, and P. Gage, “A genetic algorithm for scheduling and decomposition of multidisciplinary design problems,” in Proceedings of the 1995 Design Engineering Technical Conferences, vol 1 of DE-Vol. 82, ASME, 1995, pp. 157–164.
2. R. Balling and J. Sobieszczanski-Sobieski, “Optimization of coupled systems: A critical overview of approaches,” in Fifth AIAA/NASA/USAF/ISSMO Symposium on Multidisciplinary Analysis and Optimization, vol. AIAA-94-4330-CP, 1994, pp. 753–774.
3. D. Bertsekas, Nonlinear Programming, Athena Scientific: Belmont, Massachusetts, 1995.
4. R. Braun and I. Kroo, “Development and application of the collaborative optimization architecture in a multidisciplinary design environment,” in Multidisciplinary Design Optimization: State of the Art, N. Alexandrov and M. Hussaini (Eds.), SIAM, 1997, pp. 98–116.
5. E. Cramer Jr., J. Dennis, P. Frank, R. Lewis, and G. Shubin, “Problem formulation for multidisciplinary optimization,” SIAM Journal on Optimization, vol. 4, pp. 754–776, 1994.
6. S. Friedberg, A. Insel, and L. Spence, Linear Algebra, Prentice Hall, N.J., 1989.
7. P. Gill, W. Murray, and M. Saunders, User’s Guide for QPOPT 1.0: A FORTRAN Package for Quadratic Programming, Aug. 1995.
8. R. Keeney and H. Raffia, Decisions with Multiple Objectives: Preference and Value Tradeoffs, Wiley: New York, 1976.
9. R. Krishnamachari and P. Papalambros, “Hierarchical decomposition synthesis in optimal system design,” Transactions of the ASME, Journal of Mechanical Design, vol. 119, no. 4, pp. 448–457, 1997.
10. R. Krishnamachari and P. Papalambros, “Optimal design of a hybrid electric powertrain system,” International Journal of Mechanics of Structures and Machines, vol. 25, no. 3, pp. 267–286, 1997.

11. I. Kroo, S. Altus, R. Braun, P. Cage, and I. Sobieski, "Multidisciplinary optimization methods for aircraft preliminary design," in Fifth AIAA/NASA/USAF/ISSMO Symposium on Multidisciplinary Analysis and Optimization, vol. AIAA-94-4325-CP, 1994, pp. 697–707.
12. A. Kusiak and J. Wang, "Decomposition of the design process," Journal of Mechanical Design, vol. 115, pp. 687–695, 1993.
13. D. Macko and Y. Haimes, "Overlapping coordination of hierarchical structures," IEEE Transactions on Systems, Man, and Cybernetics, vol. SMC-8, pp. 745–751, 1978.
14. Using MATLAB, version 5.1 edition, The Math Works, Inc.,: Natick, MA, 1997.
15. C. McCulley and C. Bloebaum, "A genetic tool for optimal design sequencing in complex engineering systems," Structural Optimization, vol. 12, pp. 186–201, 1996.
16. N. Michelena, "Optimal model decomposition of design problems," Web implementation available at http:// arc. engin. umich. edu/decomp-docs/decomp. html, 1995. UserID: ombdu, Password: ombdp.
17. N. Michelena and P. Papalambros, "Optimal model-based decomposition of powertrain system design," Transactions of the ASME, Journal of Mechanical Design, vol. 117, no. 4, pp. 499–505, 1995.
18. N. Michelena and P. Papalambros, "A hypergraph framework to optimal model-based decomposition of design problems," Computational Optimization and Applications, vol. 8, no. 2, pp. 173–196, 1997.
19. S. Nelson and P. Papalambros, "Sequentially decomposed programming," AIAA Journal, vol. 35, no. 7, pp. 1209–1216, 1997.
20. R. Nicolaides and N. Walkington, Maple: A Comprehensive Introduction, Cambridge University Press: New York, 1996.
21. J. Rogers, "A knowledge-based tool for multilevel decomposition of a complex design problem," Technical Report TP2903, NASA, 1989.
22. T. Shima, "Hierarchical overlapping coordination of large scale systems with multiple objectives," Ph.D. Thesis, Case Western Reserve University, 1982.
23. T. Shima and Y. Haimes, "The convergence properties of hierarchical overlapping coordination," IEEE Transactions on Systems, Man, and Cybernetics, vol. SMC–14, pp. 74–87, 1984.
24. J. Sobieszczanski-Sobieski, "Optimization by decomposition: A step from hierarchic to non-hierarchic systems," in Second NASA/Air Force Symposium on Recent Advances in Multidisciplinary Analysis and Optimization, Hampton, VA, Sept. 1988. NASA CP–3031, Part 1. Also NASA TM–101494.
25. J. Sobieszczanski-Sobieski and R. Haftka, "Multidisciplinary aerospace design optimization: Survey of recent developments," Structural Optimization, vol. 14, pp. 1–23, 1997.
26. D. Steward, Systems Analysis and Management: Structure, Strategy, and Design. Petrocelli Books: Princeton, NJ, 1981.
27. T. Wagner and P. Papalambros, "A general framework for decomposition analysis in optimal design," in Advances in Design Automation, vol. 2, B.J. Gilmore (Ed.), ASME: New York, 1993, pp. 315–322.