Enhanced Collaborative Optimization:
Application to an Analytic Test Problem and Aircraft Design

Brian Roth* and Ilan Kroo†
Stanford University, Stanford, California, 94305, USA

This paper provides an introduction to a new method for distributed optimization based on collaborative optimization, a decomposition-based method for the optimization of complex multidisciplinary designs. The key idea in this approach is to include models of the global objective and all of the subspace constraints in each subspace optimization problem while maintaining the low dimensionality of the system level (coordination) problem. Results from an analytic test case and an aircraft family design problem suggest that the new approach is robust and leads to a substantial reduction in computational effort.

I. Introduction

Collaborative optimization (CO) is a method for the design of complex, multidisciplinary systems that was originally proposed[1] in 1994. CO is one of several decomposition-based methods that divides a design problem along disciplinary (or other convenient) boundaries. The idea is to mirror the natural divisions found in aerospace design companies. In these settings, engineers are often divided into design groups by disciplinary expertise. Disciplinary analysis tools tend to be complex in nature, and it is often impractical to integrate multiple analysis codes for the purpose of multidisciplinary optimization. Rather, CO offers a means of coordinating separate analyses, even leveraging discipline-specific optimization techniques. Relative to other decomposition-based methods, CO provides the disciplinary subspaces with an unusually high level of autonomy. This enhances their ability to independently make design decisions pertinent primarily to their discipline.

Collaborative optimization has been successfully applied to a variety of mathematical test problems and practical engineering design problems. For example, it has been used for the conceptual design of launch vehicles,[2,3] high speed civil transports[4] and unmanned aerial vehicles.[5] However, it also suffers from some challenges, as documented by Alexandrov,[6,7] DiMiguel,[8] and others. DeMiguel highlights features of CO that have an adverse effect on robustness and computational efficiency. Three of these deficiencies are briefly reviewed in the following paragraph, since they strongly motivated the development of enhanced collaborative optimization (ECO).

The basic CO formulation is composed of two levels. The system level (top level) is given by Equation 1. Note that the variable set ($\mathbf{z}$) includes only those variables required by more than one subspace. The $\mathbf{x}_s^*$ are subspace target responses that provide each subspace's best attempt to meet the system level targets ($\mathbf{z}$). The $\mathbf{x}_s^*$ are treated as dependent variables, which means that the subspaces must be re-optimized each time that the system level evaluates its constraints.

$$
\begin{aligned}
\min_{\mathbf{z}} \quad & F && (1)\\
\text{subject to} \quad & J_i = \|\mathbf{z} - \mathbf{x}_\mathbf{s}^*\|_2^2 \leq 0, \quad i = 1, \ldots, n \\
\text{where} \quad & F(\mathbf{z}) \text{ is the global objective} \\
& \mathbf{z} \text{ are variables (i.e., system level targets for shared variables)} \\
& \mathbf{x}_s^* \text{ are dependent variables (i.e., subspace target responses)} \\
& n \text{ is the number of subspaces}
\end{aligned}
$$

*Doctoral Candidate, Department of Aeronautics and Astronautics, AIAA Student Member
†Professor, Department of Aeronautics and Astronautics, AIAA Fellow

The subspace level (lower level) is illustrated in Equation 2. The subspace objective focuses entirely on achieving compatibility by seeking to match targets for shared variables that have been sent by the system level. The set of independent variables includes both "shared" (x) and "local" (x_L) variables. The set of shared variables includes both independent (x) and dependent (y) variables. The dependent variables are also referred to as "coupling" variables.

$$\begin{aligned} \min_{\mathbf{x},\mathbf{x}_L} \quad & J_i = \|\mathbf{x}_s - \mathbf{z}\|_2^2 \\ \text{subject to} \quad & \mathbf{g}(\mathbf{x},\mathbf{x}_L) \geq 0 \\ \text{analysis} \quad & \mathbf{y} = \mathbf{y}\left(\mathbf{x},\mathbf{x}_L\right) \\ \text{where} \quad & \mathbf{x} \text{ are independent shared variables} \\ & \mathbf{x}_L \text{ are local variables (variables relevant only to the local subspace)} \\ & \mathbf{y} \text{ are dependent shared variables (or "coupling" variables)} \\ & \mathbf{x}_s = [\mathbf{x},\mathbf{y}] \text{ are shared variables (local copies of system level variables)} \\ & \mathbf{z} \text{ are parameters (targets from the system level)} \\ & \mathbf{g}(\mathbf{x},\mathbf{x}_L) \text{ are local constraints} \end{aligned} \tag{2}$$

Briefly consider three difficulties[8] with the bi-level optimization problem stated in Equations 1 and 2.

1. The system level Jacobian is singular at the solution. This can be seen by noting that the constraint gradients are given by $\nabla J = 2.0(\mathbf{z} - \mathbf{x}_s^*)$. Even with a robust optimizer, this has an adverse impact on the rate of convergence.

2. The Lagrange multipliers in the subspace problems are either zero or converge to zero as $\mathbf{z}$ converges to $\mathbf{z}^*$. This can adversely effect subspace convergence.

3. The subspace responses (J_i) are, in general, nonsmooth functions of the targets, $\mathbf{z}$. As a result, the system level constraints are nonsmooth, hindering local and global convergence proofs for the system level problem.

In response to these performance challenges, several significant updates to the original CO framework have been proposed. The following paragraphs highlight two of these methods.

Sobieski proposed the use of response surfaces within the collaborative optimization framework.[9] Sobieski's key idea was to replace the system level constraints with response surface models of the subspace optimal responses. This is in contrast to modeling the subspaces themselves, using a response surface. Additional details can be found in his thesis[10] and AIAA Journal article.[9] Sobieski's method resolves many (though not all) of the computational challenges associated with CO. First, while the system level Jacobian is still singular at the solution, the subspace responses (as represented by response surfaces) are inexpensive to evaluate. Hence, slow system level convergence no longer adversely impacts computational efficiency. Second, the subspace Lagrange multipliers still converge to zero. Thus, subspace efficiency is still adversely affected. Third, the response surfaces provide smooth subspace responses, yielding smooth system level constraints. However, representing a non-smooth function with a smooth surface may not be an ideal approach.

Modified collaborative optimization (MCO) takes a different approach to addressing the computational issues associated with CO. DiMiguel proposed two related methods that lead to improved efficiency in the subspace and system level problems. Both formulations have unconstrained system level problems, eliminating CO's singular Jacobian. The subspace problems have also been reformulated to ensure that the Lagrange multipliers do not converge to zero. However, the subspace responses ($\mathbf{x}_s^*$) are still non-smooth functions of the system level targets. Additional details can be found in DeMiguel's thesis[11] and AIAA conference paper.[8]

In fact, the subspace responses are fundamentally non-smooth functions of the system level targets. This non-smoothness is caused by changes in the active set of subspace constraints. The key idea in ECO is to communicate the source of non-smoothness by sharing models of each subspace's constraints with all other subspaces. This has the added benefit that it prevents subspaces from engaging in a tug-of-war struggle over appropriate choices for shared design variables. With careful formulation, constraint models can be shared without increasing the dimensionality of the system level optimization problem. The ECO method leads to a reformulation of both the subspace and system level problems, eliminating the ill-conditioning associated with the original CO formulation.

ECO also seeks to enhance subspace design authority. One of the strengths of the original CO method is that subspaces have exclusive control over local design decisions (i.e., subspace-specific variables). However, the subspace objective focuses exclusively on satisfying compatibility rather than directly reducing the global objective. In contrast, in the design of a complex aerospace system, the aerodynamics group would expect to work toward minimizing drag rather than trying to best match some set of targets. So, it seems preferable to enable the subspaces to work directly on relevant portions of the global objective. This idea has been incorporated into ECO, as described in the next section.

This paper is organized as follows. Section II-A provides an overview of the method. Sections II-B and II-C offer a detailed description of each level of the bi-level problem. Sections III and IV illustrate the application of ECO to an analytic test case and an aircraft design problem.

II. Description of the Method

This section provides an overview of enhanced collaborative optimization (ECO), a new method based on collaborative optimization (CO). The first subsection provides a high-level introduction to the method. The second and third subsections provide a precise, generalized, description of the method. The last subsection describes the solution process.

A. High-level Introduction

The system level is an unconstrained minimization problem. The objective is to ensure that all subspaces use the same values of shared variables while satisfying their local constraints. Note that the global objective (i.e., the overarching design goal) is not present in the system level objective. The system level's entire goal is to achieve compatibility between subspaces. The following is a simplified definition of the system level.

$$\begin{array}{rl}
\min\limits_{\mathbf{z}} & J_{sys} = \sum (\mathbf{z} - \mathbf{x}^*)^2 \\
\text{subject to} & \text{No constraints} \\
\text{where} & \mathbf{z} \text{ are the system level targets (i.e., suggestions) for shared variables} \\
& \mathbf{x}^* \text{ represent each subspace's best attempt to match the system level targets,} \\
& \qquad \text{subject to local constraints}
\end{array}$$

The subspaces are responsible for most of the design decisions. Their objective function includes three components: a quadratic model of the global objective, a quadratic measure of compatibility, and a set of slack variables. Their constraint set includes local constraints and models of constraints from other subspaces. The i^{th} subspace is defined as follows.

$$\begin{array}{rl}
\min\limits_{\bar{\mathbf{x}}=[\mathbf{x},\mathbf{x}_L]} & J_i = \widetilde{F}(\mathbf{x}_s) + \lambda_C \sum (\mathbf{x}_s - \mathbf{z})^2 + \lambda_F \sum \mathbf{s} \\
\text{subject to} & \mathbf{g}^{(i)}(\mathbf{x}_s, \mathbf{x}_L) \geq 0 \\
& \widetilde{\mathbf{g}}^{(j)}(\mathbf{x}_s) + \mathbf{s}^{(j)} \geq 0, \quad j = 1..n, \quad j \neq i \\
& \mathbf{s} \geq 0 \\
\text{where} & \mathbf{x} \text{ are independent shared variables} \\
& \mathbf{x}_L \text{ are local variables (i.e., relevant only to the current subspace)} \\
& \mathbf{y} \text{ are dependent shared variables (or ``coupling'' variables)} \\
& \mathbf{x}_s = [\mathbf{x}, \mathbf{y}] \text{ are shared variables (i.e., relevant to multiple subspaces)} \\
& \mathbf{s} \text{ are slack variables, which ensure a feasible subspace problem} \\
& \mathbf{z} \text{ are parameters, provided by the system level, that act as targets} \\
& \lambda_C \text{ is a compatibility penalty parameter} \\
& \lambda_F \text{ is a feasibility penalty parameter} \\
& \widetilde{F} \text{ is a quadratic model of the global objective} \\
& \mathbf{g}^{(i)} \text{ are local constraints in subspace } i \\
& \widetilde{\mathbf{g}}^{(\mathbf{j})} \text{ are linear models of the constraints in subspace } j
\end{array}$$

B. System Level Problem

This section provides additional details on the intended function and practical implementation of the system level problem. The notation adopted in sections B and C precisely defines the most general system level and subspace problems. The system level is defined as:

$$\begin{aligned} \min_{\mathbf{z}} \quad & J_{sys} = \sum_{i=1}^{n} \sum_{j=1}^{n_{s_i}} (z_j - x_{s_j}^{*(i)})^2 \\ \text{subject to} \quad & \text{No constraints} \\ \text{where} \quad & n \text{ is the number of subspaces} \\ & n_{s_i} \text{ is the number of shared variables in subspace } i \end{aligned} \tag{3}$$

The original version of collaborative optimization (CO) provides a significant degree of independence for each disciplinary subspace. This enables disciplinary experts to run their own codes using discipline-preferred optimization techniques. However, each subspace has very limited knowledge of the actions and preferences of the other subspaces. Information is only shared indirectly through the system level targets. As a result, the system level must retain responsibility for selecting the shared variables. In contract, ECO provides each subspace with a direct understanding of the other subspaces' preferences (i.e., constraints). This enables the transfer of most of the system level decision-making process to the individual subspaces. (The subspaces direct the system level optimization process through target responses.) The system level coordination task is now limited to providing dynamic "move limits," which prevent the subspaces from taking large steps in the wrong direction based on limited (i.e., linearly approximated) information from the other subspaces.

Two distinct approaches to the system level problem are possible. Both approaches have the same functional form, as detailed in equation 3. However, they differ in their treatment of the $\mathbf{x}_s^*$. The first method assumes that the $\mathbf{x}_s^*$ are a function of $\mathbf{z}$. This requires that the subspace optimization problems are called (to update $\mathbf{x}_s^*$) each time that the system level objective is evaluated. (This process is equivalent to the one that is used in the original version of CO.[3]) The second method eliminates the need for repeated calls to the subspaces during the solution of the system level problem. In specific, the $\mathbf{x}_s^*$ are treated as parameters in the system level problem. The process proceeds as follows.

The system level sends an initial target set ($\mathbf{z}$) to the constraint modeling subroutines and then to the subspaces. The subspaces solve for the $\mathbf{x}_s^*$ and return them to the system level. The system level treats the $\mathbf{x}_s^*$ as parameters (rather than functions of $\mathbf{z}$), allowing it to solve its optimization problem without further calls to the subspaces. The updated target set ($\mathbf{z}$) is sent to the constraint modeling subroutines and subspaces. This process is repeated until compatibility is achieved. Note that a similar iterative process has successfully been implemented for a variety of practical problems using analytical target cascading (ATC).[12–16] In fact, since the system level problem is unconstrained, the system level optimum is simply the average of the target responses returned from the subspaces. Both methods appear promising, and both have successfully solved a suit of test cases. The remainder of this paper focuses on the use of simple averaging at the system level.

C. Subspace Problem

This section explores the subspace formulation in detail. First, it describes the subspace optimization problem. Second, it presents the development of subspace constraint models. Third, it introduces a method for modeling local subspace analyses. Fourth, it discusses appropriate selection of the penalty parameters λ_F and λ_C.

1. Subspace Description

The i^{th} subspace is defined as:

$$
\begin{aligned}
\min_{\bar{\mathbf{x}}=[\mathbf{x_s},\mathbf{x_L},\mathbf{s_g},\mathbf{s_h},\mathbf{e}]} \quad & J_i = \widetilde{F}(\mathbf{x}_s) + \lambda_C \sum_{k=1}^{n_{s_i}} (x_{s_k} - z_k)^2 \\
& + \lambda_F \sum_{j=1}^{n} \sum_{k=1}^{n_{g_j}} s_{g_k}^{(j)} + \lambda_F \sum_{j=1}^{n} \sum_{k=1}^{n_{h_j}} \left(s_{h_k}^{(j)} + e_k^{(j)} \right), \quad j \neq i \qquad (4) \\
\text{subject to} \quad & g_k^{(i)}(\mathbf{x}_s, \mathbf{x}_L) \geq \ 0, \quad k = 1..n_{g_i} \\
& h_k^{(i)}(\mathbf{x}_s, \mathbf{x}_L) = 0, \quad k = 1..n_{h_i} \\
& \widetilde{g}_k^{(j)}(\mathbf{x}_s) + s_{g_k}^{(j)} \geq \ 0, \quad j = 1..n, \quad k = 1..n_{g_j}, \quad j \neq i \\
& \widetilde{h}_k^{(j)}(\mathbf{x}_s) + s_{h_k}^{(j)} - e_k^{(j)} = 0, \quad j = 1..n, \quad k = 1..n_{h_j}, \quad j \neq i \\
& \mathbf{s_g}, \mathbf{s_h}, \mathbf{e} \geq 0 \\
\text{where} \quad & \widetilde{F} \text{ is a quadratic model of the global objective} \\
& n_{g_j} \text{ is the number of inequality constraints in subspace } j \\
& n_{h_j} \text{ is the number of equality constraints in subspace } j \\
& \widetilde{g}_k^{(j)} \text{ is a linear model of the } k_{th} \text{ inequality constraint in subspace } j \\
& \widetilde{h}_k^{(j)} \text{ is a linear model of the } k_{th} \text{ equality constraint in subspace } j
\end{aligned}
$$

The subspace receives targets ($\mathbf{z}$) and constraint model coefficients ($\partial \mathbf{g}^{(j)} / \partial \mathbf{z}$) from the system level, which are treated as parameters. The subspace returns target responses ($\mathbf{x}_s^*$). Note that each subspace (as illustrated by the i^{th} subspace) requires models of the constraints from all other subspaces. Though the original constraints are typically a function of both local and shared variables, the constraint models used in subspace i are functions only of the shared variables in subspace i. Consider a linear model of the k_{th} inequality constraint in subspace i. To ease the notation, the superscript (i) is dropped.

$$
\widetilde{g}_k \;=\; g_k|_{\mathbf{z},\mathbf{x_L^*}} + \left.\frac{\partial g_k}{\partial z_1}\right|_{\mathbf{z},\mathbf{x_L^*}} (x_{s_1} - z_1) + \left.\frac{\partial g_k}{\partial z_2}\right|_{\mathbf{z},\mathbf{x_L^*}} (x_{s_2} - z_2) + \ldots = g_k|_{\mathbf{z},\mathbf{x_L^*}} + \left.\frac{\partial g_k}{\partial \mathbf{z}}\right|_{\mathbf{z},\mathbf{x_L^*}} (\mathbf{x_s} - \mathbf{z})
$$

The coefficients in the constraint model are described in the next section.

2. Constraint Modeling

The goal of constraint modeling is to capture the effect of the target variables ($\mathbf{z}$) on the subspace constraints. To accomplish this task, consider the optimization sub problem shown in Equation 5.

$$
\begin{aligned}
\min_{\bar{\mathbf{x}}=[\mathbf{x}_L,\mathbf{s_g},\mathbf{s_h},\mathbf{e_h}]} \quad & J_{i_{CVM}} = \sum_{k=1}^{n_{g_i}} s_{g_k} + \sum_{k=1}^{n_{h_i}} (s_{h_k} + e_{h_k}) \qquad (5) \\
\text{subject to} \quad & \mathbf{g_k}(\mathbf{x}_s, \mathbf{x}_L) + \mathbf{s_g} \geq \ 0, \quad k = 1..n_{g_i} \\
& \mathbf{h_k}(\mathbf{x}_s, \mathbf{x}_L) + \mathbf{s_h} + \mathbf{e_h} = 0, \quad k = 1..n_{h_i} \\
& \mathbf{s_g}, \mathbf{s_h}, \mathbf{e_h} \geq 0 \\
\text{where} \quad & \mathbf{x_s} \text{ are parameters with values, } \mathbf{x_s} = \mathbf{z} \\
& \mathbf{g_k}(\mathbf{x}_s, \mathbf{x}_L) \text{ are local inequality constraints} \\
& \mathbf{h_k}(\mathbf{x}_s, \mathbf{x}_L) \text{ are local equality constraints} \\
& n_{g_i} \text{ is the number of inequality constraints in subspace i} \\
& n_{h_i} \text{ is the number of equality constraints in subspace i}
\end{aligned}
$$

The sub problem receives target values ($\mathbf{z}$), which it treats as parameters. Its sole objective is to minimize the cumulative constraint violation through optimal selection of the local variables ($\mathbf{x}_L$). This leads to its name: "constraint violation minimization" (or CVM) problem. The solution to the CVM problem provides all of the information necessary for constraint modeling. First, it provides the constant coefficient, $g_k|_{\mathbf{z},\mathbf{x}_\mathbf{L}^*}$, for each constraint model. This is obtained directly from the optimal values of the slack and excess variables. Second, it provides each linear coefficient, $\frac{\partial g_k}{\partial z_i}\Big|_{\mathbf{z},\mathbf{x}_\mathbf{L}^*}$, as illustrated in Equation 6.

$$
\begin{aligned}
\left.\frac{\partial g}{\partial z_i}\right|_{z,x_L^*} &= \left(\frac{\partial g}{\partial x_s}\right)\left(\frac{dx_s}{dz_i}\right) + \left(\frac{\partial g}{\partial x_L}\right)\left(\frac{dx_L^*}{dz_i}\right) \\
&= \left(\frac{\partial g}{\partial x_s}\right) + \left(\frac{\partial g}{\partial x_L}\right)\left(\frac{dx_L^*}{dz_i}\right) \qquad (6)
\end{aligned}
$$

where $\left(\frac{\partial g}{\partial x_s}\right)$ is obtained from differentiating the constraints

$\left(\frac{\partial g}{\partial x_L}\right)$ is obtained from differentiating the constraints

$\left(\frac{dx_L^*}{dz_i}\right)$ is obtained from 2nd-order post-optimality sensitivity analysis[17]

Note that this approach to constraint modeling indirectly captures the effect of the local variables without sharing them outside of the subspace. The last term requires solution of the following set of linear equations,[17] where the set of independent variables is $\mathbf{x} = [\mathbf{x_L}, \mathbf{s_g}, \mathbf{s_h}, \mathbf{e_h}]$.

$$
\begin{bmatrix} \nabla^2_{xx}\mathscr{L}^* & -(A^*)^T \\ A^* & 0 \end{bmatrix} \begin{bmatrix} \frac{dx}{dz_i} \\ \frac{d\lambda^*}{dz_i} \end{bmatrix} = \begin{bmatrix} -\nabla^2_{z_i x}\mathscr{L}^* \\ -\frac{\partial c^*}{\partial z_i} \end{bmatrix} \qquad (7)
$$

3. Modeling Subspace Analyses

The preceding paragraphs have focused on developing models of the subspace constraints that can be shared with other subspaces. Since local analyses also effect the solution of the subspace optimization problem, it is important to model these analyses in other subspaces. This is accomplished using the same process outlined for constraints.[18]

4. Penalty Parameters

The subspace objective is a combination of three terms: (1) a quadratic model of the global objective, (2) a compatibility term, and (3) a constraint violation term. Terms (2) and (3) are accompanied by penalty parameters, λ_C and λ_F. The following is a brief description of the rationale for selecting λ_F and λ_C. λ_C determines the compromise between exploration (small λ_C) and exploitation (large λ_C). While its selection impacts ECO's computational efficiency, convergence should be obtained for a reasonable range of values. λ_F determines the relative weight that is placed on satisfying the constraint models. Since the constraint models are only linear approximations of the original constraints, it is unwise for the subspaces to venture too far from the design targets ($\mathbf{z}$) while seeking to satisfy the constraint models. For the test cases investigated in this paper and in Reference 18, convergence was obtained for a reasonable range of values. Specific details are provided for each test case.

The compatibility term warrants an additional note. Unlike the original version of CO, the compatibility term in the subspace objective does NOT ensure compatibility. As with any quadratic penalty function, it will only be precisely satisfied in the limit as $\lambda_C \to \infty$. Rather, it acts as a dynamic "move limit," guiding the optimization process. This "guide" is needed since each subspace has only limited knowledge of the other subspaces' constraints. Without the "guide," this limited knowledge might be wrongly exploited.

D. Solution Process

This section describes the solution process for ECO (Enhanced Collaborative Optimization). ECO requires a two-step process to build constraint models and solve the subspace optimization problems. In step one, the system level calls a set of subroutines that construct constraint models. (One subroutine corresponds to each subspace.) These model-builders typically take the form of constraint violation minization (CVM) problems, as detailed in section C. The system level sends the current targets ($\mathbf{z}$). The model-builders return sets of "constraint model coefficients" that can be used to construct linear models of all subspace constraints, linearized about $\mathbf{z}$. In step two, the system level calls all subspaces. The system level sends targets, $\mathbf{z}$, and constraint model coefficients. The subspaces return target responses, $\mathbf{x}_s^*$. The system level problem is then solved, and the targets ($\mathbf{z}$) are updated. The process is repeated until compatibility is achieved.

III. Analytic Test Case

This section illustrates the application of ECO to an analytic test case. Results are compared with those obtained via the original version of CO. Additional test cases can be found in [18]. The problem explored in this section was introduced by Sellar, Batill and Renaud at Notre Dame in the context of concurrent subspace optimization (CSSO).[19] Others adopted the problem, using it to compare the performance of CO, CSSO, and BLISS.[20,21] Renaud[20] concluded that CO was "ineffective and unreliable." Chen[21] concluded that while CO was easy to implement, it suffered from slow convergence. Neither paper provided a table that listed convergence details for a selection of starting points. However, both papers suggested that convergence required approximately 2000 function evaluations. Both papers based their comments on an implementation that used optimizers from the Matlab toolbox. As illustrated in this section, an SNOPT implementation is more efficient and reliable. In addition, ECO provides an additional computational savings of more than 90%. It should be noted, however, that this problem is not an ideal test case for CO because it has no local (non-shared) variables in the subspaces. Collaborative optimization is best suited to problems where local variables far outnumber shared variables in each of the subspaces.

The test case is defined as follows:

$$
\begin{aligned}
\min_{\bar{\mathbf{x}}=[x_1,x_2,x_3]} \quad & f = x_2^2 + x_3 + y_1 + e^{-y_2} \\
\text{subject to} \quad & (y_1/y_{1a}) - 1 \geq \ 0 \\
& 1 - (y_2/y_{2a}) \geq \ 0 \\
\text{Bounds} \quad & -10 \leq \ \mathbf{x} \leq \ 10 \\
\text{Analysis} \quad & y_1 = x_1^2 + x_2 + x_3 - 0.2 y_2 \\
& y_2 = y_1^{1/2} + x_1 + x_3 \\
\text{Parameters} \quad & y_{1a} = 8 \\
& y_{2a} = 24 \\
\text{Global Optimum} \quad & [x_1, x_2, x_3, y_1, y_2] = [3.0284, \quad 0.0000, \quad 0.0000, \quad 8.0000, \quad 5.8569] \\
& f = 8.00286 \\
\text{Local Minima} \quad & [x_1, x_2, x_3, y_1, y_2] = [-2.8014, \quad 0.0757, \quad 0.1021, \quad 8.0000, \quad 0.1292] \\
& f = 8.9867
\end{aligned}
$$

The solution to this integrated problem is shown in Table 1. While just four samples are shown, the optimization process converged for all starting points. Convergence (from 32 starting points) required an average of 49 function calls.

Table 1. Solution Via Integrated Approach

Starting Point	Type of Minima	System Level Iterations	Avg Subspace Iterations	Function Evaluations
$\mathbf{z} = [-10, 0, 5]$	Local	57	—	57
$\mathbf{z} = [-1, 5, 0]$	Local	55	—	55
$\mathbf{z} = [1, 0, 5]$	Global	54	—	54
$\mathbf{z} = [10, 5, 10]$	Global	62	—	62

1. CO Formulation

The test case was also solved via the original version of CO. The problem decomposition is shown below, and results are provided in Table 2.
System-level problem:

$$\begin{aligned}
\min_{\bar{\mathbf{x}}=[z_1,z_2,z_3,y_1,y_2]} \quad & f = x_2^2 + x_3 + y_1 + e^{-y_2} \\
\text{subject to} \quad & (z_1 - x_1^{(1)})^2 + (z_2 - x_2^{(1)})^2 + (z_3 - x_3^{(1)})^2 + (y_1 - y_1^{(1)})^2 + (y_2 - y_2^{(1)})^2 \leq 0 \\
& (z_1 - x_1^{(2)})^2 + (z_2 - x_2^{(2)})^2 + (z_3 - x_3^{(2)})^2 + (y_1 - y_1^{(2)})^2 + (y_2 - y_2^{(2)})^2 \leq 0
\end{aligned}$$

Subspace 1 is defined as:

$$\begin{aligned}
\min_{\bar{\mathbf{x}}=[x_1,x_2,x_3,y_2]} \quad & f_1 = (x_1 - z_1)^2 + (x_2 - z_2)^2 + (x_3 - z_3)^2 + (y_1 - yz_1)^2 + (y_2 - yz_2)^2 \\
\text{subject to} \quad & (y_1/y_{1a}) - 1 \geq 0 \\
\text{Analysis} \quad & y_1 = x_1^2 + x_2 + x_3 - 0.2y_2
\end{aligned}$$

Subspace 2 is defined as:

$$\begin{aligned}
\min_{\bar{\mathbf{x}}=[x_1,x_3,y_1]} \quad & f_2 = (x_1 - z_1)^2 + (x_2 - z_2)^2 + (x_3 - z_3)^2 + (y_1 - yz_1)^2 + (y_2 - yz_2)^2 \\
\text{subject to} \quad & 1 - (y_2/y_{2a}) \geq 0 \\
\text{Analysis} \quad & y_2 = y_1^{1/2} + x_1 + x_3
\end{aligned}$$

The solution to this problem is shown in Table 2. Convergence was achieved for a wide range of starting points, four of which are shown in the table. Convergence required an average of 598 subspace objective function evaluations (based on 32 starting points). These results are in contrast to published reports that suggested inconsistent convergence and/or thousands of subspace evaluations required for convergence.[20,21]

Table 2. Solution Via CO

Starting Point	Type of Minima	System Level Iterations	Avg Subspace Iterations	Function Evaluations
$\mathbf{z} = [-10, 0, 5]$	Local	97	6.70	650
$\mathbf{z} = [-1, 5, 0]$	Global	79	8.24	651
$\mathbf{z} = [1, 0, 5]$	Global	42	7.60	319
$\mathbf{z} = [10, 5, 10]$	Global	73	8.36	610

2. ECO Formulation

The test case has been successfully solved using ECO. The system level is defined as follows.

$$\begin{aligned}
\min_{z_1,z_2,z_3,z_4,z_5} \quad & J_{system} = \left[(z_1 - x_1^{(1)})^2 + (z_2 - x_2^{(1)})^2 + (z_3 - x_3^{(1)})^2 + (z_4 - x_4^{(1)})^2 + (z_5 - x_5^{(1)})^2\right] \\
& \quad + \left[(z_1 - x_1^{(2)})^2 + (z_3 - x_3^{(2)})^2 + (z_4 - x_4^{(2)})^2 + (z_5 - x_5^{(2)})^2\right] \\
\text{subject to} \quad & \text{No Constraints}
\end{aligned}$$

Subspace 1 is given by:

$$
\begin{aligned}
\min_{\mathbf{x}} \quad & J_1 = \left[x_2^2 + x_3 + x_4 + e^{-x_5}\right] \\
& \quad + \lambda_C \left[(x_1 - z_z)^2 + (x_2 - z_2)^2 + (x_3 - z_3)^2 + (x_4 - z_4)^2 + (x_5 - z_5)^2\right] + \lambda_F \left[e_1\right] \\
\text{where} \quad & \mathbf{x} = [x_1, x_2, x_3, x_4, x_5, e_1, e_2] \\
\text{subject to} \quad & g_1 = (x_4/y_{1a}) - 1.0 \geq 0 \\
& h_1 = x_4 - y_1 = 0 \\
& \widetilde{g}_2 = g_2(z) + \sum_{i=1}^{5} \left[\left(\frac{dg_2}{dx_i} \right) (x_i - z_i) \right] + e_1 \geq 0, \quad i \neq 2 \\
& \widetilde{h}_2 = h_2(z) + \sum_{i=1}^{5} \left[\left(\frac{dh_2}{dx_i} \right) (x_i - z_i) \right] + e_2 \geq 0, \quad i \neq 2 \\
\text{Analysis:} \quad & y_1 = x_1^2 + x_2 + x_3 - 0.2x_5
\end{aligned}
$$

Subspace 2 is given by:

$$
\begin{aligned}
\min_{\mathbf{x}} \quad & J_2 = \left[x_3 + x_4 + e^{-x_5}\right] + \lambda_C \left[(x_1 - z_z)^2 + (x_3 - z_3)^2 + (x_4 - z_4)^2 + (x_5 - z_5)^2\right] + \lambda_F \left[e_1 + e_2\right] \\
\text{where} \quad & \mathbf{x} = [x_1, x_3, x_4, x_5, e_1, e_2] \\
\text{subject to} \quad & g_2 = 1.0 - (x_5/y_{2a}) \geq 0 \\
& h_2 = x_5 - y_2 = 0 \\
& \widetilde{g}_1 = g_1(z) + \sum_{i=1}^{5} \left[\left(\frac{dg_1}{dx_i} \right) (x_i - z_i) \right] + e_1 \geq 0, \quad i \neq 2 \\
& \widetilde{h}_1 = h_1(z) + \sum_{i=1}^{5} \left[\left(\frac{dh_1}{dx_i} \right) (x_i - z_i) \right] + e_2 \geq 0, \quad i \neq 2 \\
\text{Analysis:} \quad & y_2 = x_4^{1/2} + x_1 + x_3
\end{aligned}
$$

Consider constructing a model for h_2. Since y_2 is a local, dependent variable, it cannot be directly included in the model. Rather, finite differencing (or analytic differentiation, if available) should be used to construct a linear model of the analysis response, y_2, as a function of the shared variables, x_1, x_3, x_4, and x_5. This yields $\widetilde{h}_2 = h_2(z) + 1.0(x_5 - z_5) - \left[1.0(x_1 - z_1) + 1.0(x_3 - z_3) + 0.5z_4^{-0.5}(x_4 - z_4)\right]$.

The solution to the test case using ECO is shown in Table 3. Results were generated using $\lambda_F = 10$ and $\lambda_C = 0.1$. Convergence required an average of 58.8 subspace objective function evaluations (based on 32 starting points). This highlights an average computational savings of more than 90% over the original version of CO. Note that the ECO subspace problems require 40% fewer iterations for convergence. This highlights the fact that the ECO subspace problems no longer suffer from the ill-conditioning inherent in the original version of CO. In summary, enhanced collaborative optimization (ECO) consistently and efficiently solves this test case.

Table 3. Solution Via ECO

Starting Point	Type of Minima	System Level Iterations	Avg Subspace Iterations	Function Evaluations
$\mathbf{z} = [-10, 0, 5]$	Global	12	5.1	61
$\mathbf{z} = [-1, 5, 0]$	Global	15	4.3	65
$\mathbf{z} = [1, 0, 5]$	Global	7	4.7	33
$\mathbf{z} = [10, 5, 10]$	Global	15	4.3	64

IV. Design of an Aircraft Family

A product family is a set of individual products that share common components or subsystems and address a related set of market applications.[22] In an aerospace context, a product family is usually comprised of a baseline aircraft and its derivatives or variants. A typical approach is to design the family members sequentially, beginning with the baseline design. In this approach, shared components are designed primarily for the needs of the baseline aircraft. An alternative is to design all family members at the same time.[23] This yields shared components that are non-optimal from the perspective of individual family members, but are family-optimal. In other words, designing all family members at the same time yields common components that provide an optimal compromise between the competing needs of all family members.

In this paper we consider a family composed of two aircraft. The aircraft share a common main wing section, and are allowed to have unique wing tip extensions. The objective function includes cost measures that distinguish between unique aircraft and families of aircraft, namely, a detailed model of acquisition cost and a reasonable estimate of fuel cost.

The acquisition cost model used in this paper is based on recent work by Markish.[24] Acquisition cost is split into manufacturing and development costs. A manufacturing learning curve is applied such that cost decreases with the number of units produced. For example, the 100th unit costs less to manufacture than the 1st unit. Development cost is non-recurring and is averaged over the total number of aircraft produced. For every part of a new aircraft design that has already been developed for another aircraft (i.e., for another aircraft in the family), the non-recurring cost is significantly lower. Thus, the effects of commonality are captured by the acquisition cost model.

Many airline labor costs, such as pension plans, are relatively unaffected by an airline's choice of aircraft fleet. Other labor costs, however, such as crew scheduling, training, and maintenance, are significantly impacted by choice of aircraft fleet.[25] These costs are difficult to model and have not been included in the present cost model. Though not specifically addressed in this paper, product families provide a potentially significant benefit in this area.

Aircraft performance is evaluated using the Program for Aircraft Synthesis Studies (PASS), an aircraft conceptual design tool based on a collection of McDonnell-Douglas methods, DATCOM correlations, and new analyses developed specifically for conceptual design and performance. PASS has evolved over more than 15 years.[26] A detailed description of these methods may be found on the website of an aircraft design course at Stanford University.[27]

While existing conceptual design tools such as PASS are well-suited for the design of individual aircraft, a more detailed structural model is required for aircraft family design. For example, wing weight is computed using the following semi-empirical equation.

$$W_{wing} = 4.22 S_{wg} + 1.642 * 10^{-6} \frac{N_{ult} b^3 \sqrt{W_{TO} W_{ZFW}} (1 + 2\lambda)}{(t/c)_{avg} cos^2(\Lambda_{ea}) S_{wg} (1 + \lambda)} \tag{8}$$

Note that wing weight is a function of wing geometry (S_{wg}, b, λ, etc.) as well as aircraft weight (W_{TO}). Thus, sharing a common wing geometry is not sufficient to ensure wing commonality. An additional issue is the need to compute the weight of individual wing sections such as root and tip extensions. These issues associated with wing commonality suggest the need for a more detailed wing weight model. While a finite element model was an option, the goal was a low-fidelity model consistent with existing conceptual design tools that captured the desired effects and was computationally efficient. The solution was a simple wing-box model in which the wing skin carried the bending load. An analysis estimated the load distribution on the wing and computed the material necessary to resist the resulting bending moment. Since high-lift systems, control surfaces, and minimum gauge material add to the final wing weight, a new equation was developed based on "bending material" and correlated to existing aircraft. This equation is listed below, where W_{str} is the weight of material needed to resist bending, W_{min} is the weight of minimum gauge material, and S_{wing} is the wing area

$$W_{wing} = 1.35(W_{str} - W_{min}) + 4.9 S_{wing}. \tag{9}$$

Given a wing weight equation appropriate for modeling commonality between family members, the next step was to identify an appropriate means of parameterizing the wing for use in a decomposed optimization problem. The goal was to minimize the dimensionality while ensuring commonality. It was noted that an approximately quadratic relationship exists between skin thickness and spanwise location in the simple wing model. This enabled a three-term parameterization, where the skin thickness was defined at the following spanwise locations: wing root (T_1), 33% span (T_2), and 67% span (T_3) (of the main wing section). The wing tip was intentionally avoided in this parameterization since it is often sized by minimum gauge requirements rather than stress constraints. This yielded the following set of eight variables that uniquely define the main wing section: S_{wing}, AR_{wing}, λ, Λ, (t/c), T_1, T_2, and T_3. (Note that the current investigation focuses on commonality of the main wing section, with each aircraft allowed to have a unique wing tip extension. Future work will include the capability for wing root and wing tip extensions.)

A. Problem Statement

This paper considers an aircraft family that includes two aircraft types, A and B, designed to fulfill missions 1 and 2, respectively. Mission 1 requires a range of 3400 nautical miles (nmi) and an aircraft capacity of 300 passengers. Mission 2 requires a range of 8200 nmi and an aircraft capacity of 260 passengers. Forecasts suggest a market need for 800 type A aircraft, and a need for 400 type B aircraft. In addition to mission requirements, constraints such as balanced field length and second segment climb are included.

The design problem is decomposed using enhanced collaborative optimization. The system level design problem coordinates the design of all family members. Each subspace optimization problem is responsible for the design of one family member. Local design variables specify all portions of the aircraft not shared in common with other aircraft in the family. Component commonality in the present study is limited to the main wing. Each family member has the freedom to specify its own wing tip extension area. This decomposition by family member provides many of the same benefits afforded by the disciplinary decomposition of multidisciplinary problems. Namely, it simplifies analysis integration, it provides a means to manage problem complexity, and it enables concurrent design of all family members.

The aircraft family design problem involves 16 design variables for each of the two aircraft types. The design variables for each aircraft ($x_{1i} \ldots x_{16i},\ i \in \{A, B\}$) are described in Table 4.

Table 4. Design variables for the aircraft family design problem

Variable	Name	Description	Aircraft A Var Bounds	Aircraft B Var Bounds
x_{1i}	W_{TO}	takeoff weight	300^k - 450^k lbs	450^k - 600^k lbs
x_{2i}	Thrust	sea level static thrust	45^k - 65^k lbs	75^k - 105^k lbs
x_{3i}	X_{wing}	location of wing root LE	0.20 - 0.40	0.20 - 0.40
x_{4i}	S_h/S_{ref}	nondimen. horiz. tail area	0.20 - 0.35	0.20 - 0.35
x_{5i}	Alt_I	initial cruise altitude	32^k - 45^k ft	32^k - 45^k ft
x_{6i}	Alt_F	final cruise altitude	32^k - 45^k ft	32^k - 45^k ft
x_{7i}	$Mach$	cruise Mach number	0.75 - 0.92	0.75 - 0.92
x_{8i}	$flap_{TO}$	takeoff flap deflection	0.0 - 30.0	0.0 - 30.0
x_{9i}	S_{wt}	wing tip extension area	0 - 140 ft^2	0 - 140 ft^2
x_{10i}	S_{wing}	main wing area	2000 - 4000 ft^2	2000 - 4000 ft^2
x_{11i}	AR_{wing}	main wing aspect ratio	6.0 - 12.0	6.0 - 12.0
x_{12i}	(t/c)	thickness to chord ratio	0.08 - 0.13	0.08 - 0.13
x_{13i}	Λ	wing sweep	20.0 - 35.0	20.0 - 35.0
x_{14i}	T_1	skin thickness at wing root	0.06 - 2.5	0.06 - 2.5
x_{15i}	T_2	skin thickness at 33% span	0.06 - 2.0	0.06 - 2.0
x_{16i}	T_3	skin thickness at 67% span	0.06 - 1.5	0.06 - 1.5

The product family design problem imposes the constraint that the variables $x_{10i} \ldots x_{16i}$ are equal for each aircraft, since these pertain to the common component—the main wing. The vector of shared variables is

$$\mathbf{x}_s = [x_{10A} \ldots x_{16A}]^{\mathrm{T}} = [x_{10B} \ldots x_{16B}]^{\mathrm{T}}.$$

The local variables for aircraft A and B are

$$\mathbf{x}_{\ell A} = [x_{1A} \ldots x_{9A}]^{\mathrm{T}} \quad \text{and} \quad \mathbf{x}_{\ell B} = [x_{1B} \ldots x_{9B}]^{\mathrm{T}}.$$

The complete set of design variables for the product family design problem is

$$\mathbf{x} = [\mathbf{x}_{\ell A}^{\mathrm{T}}\ \mathbf{x}_{\ell B}^{\mathrm{T}}\ \mathbf{x}_s^{\mathrm{T}}]^{\mathrm{T}}.$$

Each aircraft must comply with a set of five performance constraints, whose numeric values are specific to the mission each aircraft is designed to fly (see Table 5).

Table 5. Design constraints for the aircraft family design problem

Constraint	Name	Description	Aircraft A	Aircraft B
g_1	$Range$	min range	3,400 nmi	8,200 nmi
g_2	$TOFL$	max takeoff field length	7,000 ft	10,000 ft
g_3	LFL	max landing field length	5,200 ft	6,000 ft
g_4	γ_2	min 2$^{\text{nd}}$ seg. climb grad	0.024	0.024
g_5	$stab$	stability requirement	≥ 0.10	≥ 0.10
g_6	$\hat{\sigma}_1$	normalized stress at wing root	≤ 0	≤ 0
g_7	$\hat{\sigma}_2$	normalized stress at 33% span	≤ 0	≤ 0
g_8	$\hat{\sigma}_3$	normalized stress at 67% span	≤ 0	≤ 0

The objective of the aircraft family design problem is to minimize a composite cost metric for the family, where the cost metric for each mission is normalized by the number of aircraft that fly each mission. The cost metric model is based on an estimate of direct and indirect operating costs[27] with specific attention given to acquisition cost.[24] The system objective function is given in Equation (10), where n_A and n_B are the number of aircraft A and B in the family, respectively, and p_A and p_B are the cost metrics for each aircraft.

$$f(\mathbf{x}) = \frac{n_A}{n_A + n_B} p_A(\mathbf{x}_{\ell A}, \mathbf{x}_s) + \frac{n_B}{n_A + n_B} p_B(\mathbf{x}_{\ell B}, \mathbf{x}_s). \tag{10}$$

B. Family Results

Table 6 describes the solution of the aircraft family design problem. As required, both aircraft meet all performance requirements. In addition, ECO successfully guides the two aircraft design subspaces to compatibility of the shared wing section. One result deserves further discussion. Note that Aircraft A prefers a larger wing tip extension than Aircraft B, even though it is the smaller of the two aircraft. At first glance, this is counterintuitive — one might expect the larger family member to have the larger wing area. The explanation is as follows. Because both aircraft share the same main wing section, Aircraft A carries excess wing structure. In specific, the wing root bending stress is 67% of the yield stress, as illustrated by $\hat{\sigma}_1$. This implies that Aircraft A can add additional span without paying a structural weight penalty. The additional span (i.e., wing tip extension) yields a higher aspect ratio, reducing induced drag during takeoff. This reduces the thrust required for takeoff, enabling smaller engines and a lower direct operating cost.

V. Closing Remarks

This paper has provided an introduction to enhanced collaborative optimization (ECO). ECO builds on existing decomposition-based methods such as collaborative optimization and analytical target cascading. The key idea in this new approach is to share models of the constraints with all disciplinary design teams while maintaining the low dimensionality of the system level (coordination) problem. Results from the analytic test case suggest that ECO yields significant computational savings, relative to collaborative optimization. The price for this computational savings is a small increase in the complexity of the method. In specific, a potentially significant amount of subspace constraint information must be shared among the disciplines. This is in contrast to CO in which the subspaces deal only with local constraints and shared variables. This paper also illustrates ECO's successfully application to an aircraft family design problem.

References

[1]Kroo, I., Altus, S., Braun, R., Gage, P., and Sobieski, I., "Multidisciplinary Optimization Methods for Aircraft Preliminary Design," 1994, AIAA paper 94-4325.

Table 6. Aircraft family design results

	Name	Shared	Aircraft A	Aircraft B
p	$Cost$		\$365	\$983
g_1	$Range$ (nmi)		3400	8200
g_2	$TOFL$ (ft)		7000	10000
g_3	LFL (ft)		3308	3515
g_4	γ_2		0.024	0.024
g_5	$stab$		0.100	0.100
g_6	$\hat{\sigma}_1$		-0.330	0.000
g_7	$\hat{\sigma}_2$		-0.240	0.000
g_8	$\hat{\sigma}_3$		0.000	0.000
x_1	W_{TO} (lbs)		$356 \cdot 10^3$	$531 \cdot 10^3$
x_2	$thrust$ (lbs)		$47.4 \cdot 10^3$	$77.8 \cdot 10^3$
x_3	X_{wing}		0.25	0.27
x_4	S_h/S_{ref}		0.20	0.20
x_5	Alt_I (ft)		$35.6 \cdot 10^3$	$32.0 \cdot 10^3$
x_6	Alt_F (ft)		$44.0 \cdot 10^3$	$41.1 \cdot 10^3$
x_7	$Mach$		0.835	0.822
x_8	$flap_{TO}$ (deg)		12.1	2.4
x_9	$S_{wt}(ft^2)$		129.8	57.4
x_{10}	$S_{wm}(ft^2)$	✓	$3.95 \cdot 10^3$	$3.95 \cdot 10^3$
x_{11}	AR_{wm}	✓	8.01	8.01
x_{12}	(t/c)	✓	0.130	0.130
x_{13}	Λ (deg)	✓	35.0	35.0
x_{14}	T_1 (in)	✓	0.95	0.95
x_{15}	T_2 (in)	✓	0.74	0.74
x_{16}	T_3 (in)	✓	0.40	0.40

[2]Braun, R., Kroo, I., and Moore, A., "Use of the Collaborative Optimization Architecture for Launch Vehicle Design," 1996, AIAA paper 96-4018.

[3]Braun, R., *Collaborative Optimization: An Architecture For Large-Scale Distributed Design*, Ph.D. thesis, Stanford University, April 1996.

[4]Manning, V., *Large-Scale Design of Supersonic Aircraft Via Collaborative Optimization*, Ph.D. thesis, Stanford University, 1999.

[5]Sobieski, I., *Multidisciplinary Design Using Collaborative Optimization*, Ph.D. thesis, Stanford University, August 1998.

[6]Alexandrov, N. and Lewis, R., "Analytical and Computational Aspects of Collaborative Optimization and Multidisciplinary Design," *AIAA Journal*, Vol. 40, No. 2, February 2002, pp. 301–309.

[7]Alexandrov, N. M. and Lewis, R., *Engineering Design Optimization*, chap. Comparative Properties of Collaborative Optimization and Other Approaches to MDO, MCB Univ. Press, Bradford, England, U.K., 1999, pp. 39–46.

[8]DeMiguel, A. and Murray, W., "An Analysis of Collaborative Optimization Methods," *8th AIAA/USAF/NASA/ISSMO Symposium on Multidisciplinary Analysis and Optimization*, Long Beach, CA, Sept. 6-8 2000.

[9]Sobieski, I. and Kroo, I., "Collaborative Optimization Using Response Surface Estimation," *AIAA Journal*, Vol. 38, No. 10, October 2000, pp. 1931–1938.

[10]Sobieski, I. and Kroo, I., "Aircraft Design Using Collaborative Optimization," *34th Aerospace Sciences Meeting, Reno, NV*, 1996, AIAA paper 96-0715.

[11]DeMiguel, V., *Two Decomposition Algorithms for Nonconvex Optimization Problems with Global Variables*, Ph.D. thesis, Stanford University, 2001.

[12]Kim, H., *Target Cascading in Optimal System Design*, Ph.D. thesis, University of Michigan, 2001.

[13]Kim, H., Michelena, N., Papalambros, P., and Jiang, T., "Target Cascading in Optimal System Design," *Journal of Mechanical Design*, Vol. 125, September 2003, pp. 474–480.

[14]Tosserams, S., Etman, L., Papalambros, P., and Rooda, J., "Augmented Lagrangian Relaxation for Analytical Target Cascading," *6th World Congress on Structural and Multidisciplinary Optimization*, May 30–June 3 2005.

[15]Allison, J., Walsh, D., Kokkolaras, M., Papalambros, P., and Cartmell, M., "Analytical Target Cascading in Aircraft Design," *44th AIAA Aerospace Sciences Meeting and Exhibit, AIAA-2006-1325*, Reno, Nevada, January 9 - 12 2006.

[16]Michelena, N., Park, H., and Papalambros, P., "Convergence Properties of Analytical Target Cascading," *AIAA Journal*, Vol. 41, No. 5, 2003, pp. 897–905.

[17]Sobieszczanski-Sobieski, J., Barthelemy, J., and Riley, K., "Sensitivity of Optimum Solutions of Problem Parameters," *AIAA Journal*, Vol. 20, No. 9, September 1982, pp. 1291–1299.

[18]Roth, B., *Aircraft Family Design Using Enhanced Collaborative Optimization*, Ph.D. thesis, Stanford University, August 2008.

[19]Sellar, R., Batill, S., and Renaud, J., "Response Surface Based, Concurrent Subspace Optimization for Multidisciplinary System Design," *34th Aerospace Sciences Meeting, Reno, NV*, 1996, AIAA paper 96-0714.

[20]Renaud, G. and Shi, G., "Evaluation and Implementation of Multidisciplinary Design Optimization Strategies," *ICAS Congress*, 2002.

[21]Chen, S., Zhang, F., and Khalid, M., "Evaluation of Three Decomposition MDO Algorithms," *ICAS Congress*, 2002.

[22]Meyer, M. and Lehnerd, A., *The Power of Product Platforms*, New York: Irwin Professional Publishing, 1997.

[23]Willcox, K. and Wakayama, S., "Simultaneous Optimization of a Multiple-Aircraft Family," *Journal of Aircraft*, Vol. 40, No. 4, July-August 2003.

[24]Markish, J., *Valuation Techniques for Commercial Aircraft Program Design*, Master's thesis, Massachusetts Institute of Technology, June 2002.

[25]http://www.jetblue.com/ar2002/difference.html, (Accessed May 25, 2005).

[26]Kroo, I., "An Interactive System for Aircraft Design and Optimization," *AIAA Aerospace Design Conference, Irvine, CA*, Feb 3-6 1992, AIAA paper 92-1190.

[27]Kroo, I., "AA241—Aircraft Design: Synthesis and Analysis," Course Notes [Online], URL: http://adg.stanford.edu/aa241/AircraftDesign.html, (Accessed August 1, 2008).