11th World Congress on Structural and Multidisciplinary Optimisation
07th -12th, June 2015, Sydney Australia

An efficient parallel coordination method for decomposition-based optimization using two duality theorems

Meng Xu[1], Georges Fadel[1], Margaret M. Wiecek[2], Paolo Guarneri[3]

[1] Clemson University, Department of Mechanical Engineering, Clemson, SC, USA, {xu7, fgeorge}@clemson.edu
[2] Clemson University, Department of Mathematical Science, Clemson, SC, USA, wmalgor@clemson.edu
[3] Technical University of Cluj-Napoca, Department of Electrical Machines and Drives, Cluj-Napoca, Romamia, paolo.guarneri@emd.utcluj.ro

Abstract
In decomposition-based optimization, coordination of sub-problems always plays an important role because it eliminates the inconsistencies between the decomposed sub-problems and drives their solutions towards the optimal solution of the original problem. Coordination is often carried out in an iterative manner and efficiency becomes a critical issue during this process as the complexity of engineering problems continues to grow rapidly. This growing complexity increases the computational cost of iterations and thus parallel coordination methods are preferred over non-parallel methods in many cases. In this paper, based on the Alternating Direction Method of Multipliers (ADMM), a new parallel coordination method with a high efficiency is proposed. Unlike other popular coordination methods in the literature which use duality theorems just once, the proposed method uses duality theorems twice. Specifically, the ordinary duality theorem is applied initially to the original problem to generate a dual problem and then the ADMM is applied to the dual problem. The resulting method requires fewer copies of shared variables for the decomposition, which decreases the coordination effort necessary for the optimization to converge. Numerical tests are conducted on one mathematical and one engineering problem and the results show an increase in efficiency and accuracy for the new method when compared to the centralized Augmented Lagrangian Coordination (ALC), which is one of the most popular parallel coordination methods. Additionally, this increase in performance is consistently displayed by the new method when solving a multimodal structural optimization problem repeatedly starting from different random initial designs, while the centralized ALC fails to show similar robustness.
Keywords: Decomposition-based optimization, ADMM, ALC, duality theorem

1. Introduction

The optimization task applied to modern complex systems normally involves several different disciplines and many components. This multiplicity makes it impossible and undesirable for the optimization to be conducted by one design person or one disciplinary group. As a result, the original optimization problem is usually decomposed into multiple smaller sub-problems according to certain rules (component, disciplines, model-based or hybrid) [1]–[4]. A coordination strategy is then established to deal with the couplings between these sub-problems and drive the solutions of all the sub-problems towards a consistent optimal solution for the original problem. This partition and coordination process is called decomposition-based optimization.

Some theoretical studies in mathematical optimization have been applied to the decomposition-based optimization process and the Alternating Direction Method of Multipliers (ADMM) is one result of these studies [3]. ADMM [5] is a powerful algorithm developed in the field of convex optimization. The algorithm solves problems of the form:

$$\begin{aligned} &\text{minimize} \quad f_1(\mathbf{v}) + f_2(\mathbf{z}) \\ &\text{subject to} \quad \mathbf{Av} + \mathbf{Bz} = \mathbf{e} \end{aligned} \tag{1}$$

The augmented Lagrangian function of this problem is

$$L_\rho(\mathbf{v},\mathbf{z},\mathbf{p}) = f_1(\mathbf{v}) + f_2(\mathbf{z}) + \mathbf{p}^T(\mathbf{Av} + \mathbf{Bz} - \mathbf{e}) + (\rho/2)\|\mathbf{Av} + \mathbf{Bz} - \mathbf{e}\|_2^2 \tag{2}$$

ADMM solves the problem in Eq. (1) through the following iterations

$$\mathbf{v}^{(k+1)} := \arg\min_{\mathbf{x}} L_\rho(\mathbf{v}, \mathbf{z}^{(k)}, \mathbf{p}^{(k)}) \tag{3}$$

$$\mathbf{z}^{(k+1)} := \arg\min_{\mathbf{z}} L_\rho(\mathbf{v}^{(k+1)}, \mathbf{z}, \mathbf{p}^{(k)}) \tag{4}$$

$$\mathbf{p}^{(k+1)} := \mathbf{p}^{(k)} + \rho(\mathbf{Av}^{(k+1)} + \mathbf{Bz}^{(k+1)} - \mathbf{e}) \tag{5}$$

In decomposed-based optimization, the Augmented Lagrangian Coordination (ALC) adopts ADMM and the resulting algorithm has become one of the most popular coordination methods in recent years [3][6]. ADMM enables ALC to decouple the sub-problems using the augmented Lagrangian relaxation and to update the dual multipliers through the method of multipliers [3]. There are three types of ALC algorithms which either include an inner loop or not: Exact (ENMOM) inner loop, inexact (INMOM) inner loop, and alternating direction (ADMOM) without inner loop. At each iteration, ENMOM and INMON first use the Block Coordinate Descent (BCD) [7] method to iterate, solving each sub-problem with fixed dual multipliers and penalty weights until they reach a consistent solution, which represents the termination of the inner loop. Then the dual multipliers and penalty weights are updated in the outer loop to prepare the whole optimization process for the next iteration. The difference between the two inner loop methods is that ENMOM uses a small fixed convergence tolerance for the inner loop whereas in INMOM, the inner loop tolerance is relatively large at the beginning and decreases as the optimization process proceeds. ADMOM however, contains no inner loop since it uses ADMM and all sub-problems are solved only once at each iteration. Many tests have proven that ADMOM is the most robust and efficient algorithm among the three ALC algorithms since it does not require the expensive computational efforts of the inner loops when the current solution is far from the optimal solution [3]. Furthermore, the ADMOM algorithm is shown to perform better than other popular coordination methods such as the Analytical Target Cascading (ATC) which is based on a quadratic penalty method and the Lagrangian duality-based coordination [6][8][9] which is based on the ordinary Lagrangian duality theorem.

Similar to ALC-ADMOM, Consensus Optimization via the Alternating Direction Method of Multipliers (CADMM) is another application of ADMM to decomposition-based optimization. At each iteration of CADMM, all sub-problems are solved only once and their solutions are collected together and used to calculate the consensus values for the shared variables, which are then used as targets for sub-problems in the next iteration. The efficacy of CADMM has been verified through several test problems [4][10][11].

In either ALC or CADMM, the ADMM is applied directly to the primal problem. In this study, inspired by the research in convex optimization [12], the ADMM is applied to the dual of the primal problem. Specifically, we first introduce copies of shared variables to the original optimization problem to get a primal problem set up for decomposition-based optimization, then apply the ordinary duality theorem to that primal problem to generate a dual problem, after which we apply ADMM to the dual problem. In the resulting algorithm, all sub-problems are independent from each other and thus can be solved in parallel. The derivations of the algorithm are presented in Section 2. Section 3 tests the proposed algorithm using both mathematical and engineering problems and compares its performance with that of the parallel ALC-ADMOM. Section 4 summarizes this study and proposes possible futures research directions.

2. Derivation

Depending on the ways an optimization problem is partitioned, the resulting partition can have either a hierarchical or a network structure. The hierarchical partition can be considered a special case of the network partition, which is in great demand since in many cases partitions in the engineering area are non-hierarchical. Multi-disciplinary optimization exemplifies network structures, and several nonhierarchical coordination methods have been proposed in this area [4][11][13][14][15][16]. This study deals with decomposed sub-problems with a network structure.

Assuming a quasi-separable problem with M potential sub-systems with a network partition structure, the optimization problem is written as:

$$\begin{aligned} \min_{\mathbf{y},\mathbf{x}_1,\ldots,\mathbf{x}_M} \quad & \sum_{j=1}^{M} f_j(\mathbf{y},\mathbf{x}_j) \\ \text{subject to} \quad & \mathbf{g}_j(\mathbf{y},\mathbf{x}_j) \le \mathbf{0} \\ & \mathbf{h}_j(\mathbf{y},\mathbf{x}_j) = \mathbf{0} \end{aligned} \tag{6}$$

where $\mathbf{y} \in R^N$ is the vector of shared variables, which can either be the linking design variables between two sub-problems or the analysis output of one sub-system which is required as input for another sub-system. $\mathbf{x}_j \in R^K$ is the vector of local design variables which appears only in the sub-problem j.

2.1. Applying the ordinary duality theorem to the primal problem

To decompose the problem in Eq. ((6) into M sub-problems, M-1 copies of $\mathbf{y}$ are first introduced into the problem:

$$\begin{aligned} \min_{\mathbf{y}_1,\ldots\mathbf{y}_M,\mathbf{x}_1,\ldots,\mathbf{x}_M} \quad & \sum_{j=1}^{M} f_j(\mathbf{y}_j,\mathbf{x}_j) \\ \text{subject to} \quad & \mathbf{g}_j(\mathbf{y}_j,\mathbf{x}_j) \le \mathbf{0} \\ & \mathbf{h}_j(\mathbf{y}_j,\mathbf{x}_j) = \mathbf{0} \\ & \mathbf{S}_1\mathbf{y}_1 + \ldots + \mathbf{S}_M\mathbf{y}_M = \mathbf{0} \\ & j = 1,\ldots,M \end{aligned} \tag{7}$$

where $\mathbf{y}$ is renamed as $\mathbf{y}_M$ and $\mathbf{y}_1, \ldots, \mathbf{y}_{M-1}$ are the *M-1* copies newly created.
The new equality constraints $\mathbf{S}_1\mathbf{y}_1 + \ldots + \mathbf{S}_M\mathbf{y}_M = \mathbf{0}$ are called primal consistency constraints, and are added to the formulation to ensure that all the copies of the shared variables have the same value. Since elements in $\mathbf{y}_j$ do not necessarily appear in every row of the consistency constraints, thus a selection matrix, $\mathbf{S}_j \in R^{P \times N}$, which is similar to those introduced by Michalek and Papalambros [17] for ATC and by Tosserams [3] for ALC, is adopted. The elements of $\mathbf{S}_j$ can be 0, 1 or -1 and each row of $\mathbf{S}_j$ must only contain one "1" or one "-1". For example, assume a partition in Figure 1 with three sub-problems in a network structure coupled through three shared variables:

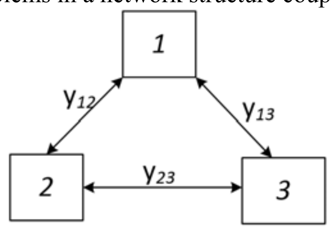


Figure 1: A partition with a three-node network structure for the illustration of selection matrix

The vector of shared variables in this partition is $\mathbf{y} = [y_{12}, y_{13}, y_{23}]^T$. After creating two copies of the shared variables $\mathbf{y}_1 = [y_{12}^1, y_{13}^1, y_{23}^1]^T$ and $\mathbf{y}_2 = [y_{12}^2, y_{13}^2, y_{23}^2]^T$, renaming $\mathbf{y}$ as $\mathbf{y}_3 = [y_{12}^3, y_{13}^3, y_{23}^3]^T$, and assigning $\mathbf{y}_1$, $\mathbf{y}_2$, $\mathbf{y}_3$ to the sub-problems 1, 2 and 3 respectively, the resulting consistency constraints for the shared variables are $y_{12}^1 - y_{12}^2 = 0$, $y_{13}^1 - y_{13}^3 = 0$, and $y_{23}^2 - y_{23}^3 = 0$. Then the selection matrixes for the three sub-problems should be $\mathbf{S}_1$ = [1 0 0;0 1 0; 0 0 0], $\mathbf{S}_2$ = [-1 0 0;0 0 0; 0 0 1] and $\mathbf{S}_3$ = [0 0 0;0 -1 0; 0 0 -1], thus $\mathbf{S}_1\mathbf{y}_1 + \mathbf{S}_2\mathbf{y}_2 + \mathbf{S}_3\mathbf{y}_3 = \mathbf{0}$ would generate the above consistency constraints. It can be seen that some elements in $\mathbf{y}_j$ are not used in sub-problem *j* at all, so in the practical implementation, $\mathbf{y}_j$ may only contain the shared variables relevant to sub-problem *j* instead of all shared variables.
Given the vector of Lagrangian multipliers $\mathbf{v} \in R^P$, the Lagrangian function *L* [7] is:

$$\begin{aligned} L(\mathbf{y}_1,\ldots\mathbf{y}_M,\mathbf{x}_1,\ldots\mathbf{x}_M,\mathbf{v}) &= \sum_{j=1}^{M} f_j(\mathbf{y}_j,\mathbf{x}_j) + \mathbf{v}^T(\mathbf{S}_1\mathbf{y}_1 +,\ldots,+\mathbf{S}_M\mathbf{y}_M) \\ &= \sum_{j=1}^{M}\left(f_j(\mathbf{y}_j,\mathbf{x}_j) + \mathbf{v}^T\mathbf{S}_j\mathbf{y}_j\right) \end{aligned} \tag{8}$$

The Lagrangian dual problem can be written as

$$\begin{aligned} &\underset{\mathbf{v}}{\text{maximize}}\ d(\mathbf{v}) \\ &\text{where}\ \ d(\mathbf{v}) = \inf_{\substack{\mathbf{y}_1,\ldots\mathbf{y}_M,\mathbf{x}_1,\ldots\mathbf{x}_M \\ \mathbf{g}_1 \le \mathbf{0},\ldots\mathbf{g}_M \le \mathbf{0},\mathbf{h}_1 = \mathbf{0},\ldots\mathbf{h}_M = \mathbf{0}}} L(\mathbf{y}_1,\ldots\mathbf{y}_M,\mathbf{x}_1,\ldots\mathbf{x}_M,\mathbf{v}) \end{aligned} \tag{9}$$

Since the Lagrangian function in Eq. (8) is separable, the dual problem can be rewritten as:

$$\begin{aligned} &\underset{\mathbf{v}}{\text{maximize}} \sum_{j=1}^{M} d_j(\mathbf{v}) \\ &\text{where}\ \ d_j(\mathbf{v}) = \inf_{\mathbf{y}_j,\mathbf{x}_j,\mathbf{g}_j \le \mathbf{0},\mathbf{h}_j = \mathbf{0}} \left(f_j(\mathbf{y}_j,\mathbf{x}_j) + \mathbf{v}^T\mathbf{S}_j\mathbf{y}_j\right) \\ &\qquad j = 1,\ldots,M \end{aligned} \tag{10}$$

2.2. Applying the ADMM duality theorem to the dual problem
At this step we apply ADMM to the dual problem in Eq. (10). *M* Copies of the Lagrangian multipliers $\mathbf{v}$ are introduced to the dual problem to decouple the objective, and new consistency constraints for the multipliers are added to the formulation. We obtain

$$\begin{aligned} &\underset{\mathbf{v}}{\text{maximize}}\ \sum_{j=1}^{M} d_j(\mathbf{z}_j) \\ &\text{subject to}\ \ \mathbf{v} - \mathbf{z}_j = \mathbf{0} \\ &\text{where}\qquad d_j(\mathbf{z}_j) = \inf_{\mathbf{y}_j,\mathbf{x}_j,\mathbf{g}_j \le \mathbf{0},\mathbf{h}_j = \mathbf{0}} \left(f_j(\mathbf{y}_j,\mathbf{x}_j) + \mathbf{z}_j{}^T\mathbf{S}_j\mathbf{y}_j\right) \\ &\qquad j = 1,\ldots,M \end{aligned} \tag{11}$$

where $\mathbf{z}_j$ is the copy of $\mathbf{v}$ at the sub-problem *j*.
Dual consistency constraints $\mathbf{v} - \mathbf{z}_j = \mathbf{0}$ are added to Eq. (11) to guarantee $\mathbf{v}$ and its *M* copies share the same value. Relaxing the dual consistency constraints for the multipliers using Augmented Lagrangian Relaxation and applying ADMM in Eqs. (3) – ((5) to the problem result in the following procedures to solve the problem in Eq. (11).

$$\mathbf{v}^{(k+1)} = \arg\min_{\mathbf{v}}\{\sum_{j=1}^{M}((\mathbf{p}_j^{(k)})^T\mathbf{v}) + \frac{\rho}{2}\sum_{j=1}^{M}\left\|\mathbf{v}-\mathbf{z}_j^{(k)}\right\|_2^2\} \tag{12}$$

$$\mathbf{z}_j^{(k+1)} = \arg\max_{\mathbf{z}_j}\{d_j(\mathbf{z}_j) + (\mathbf{p}_j^{(k)})^T\mathbf{z}_j - \frac{\rho}{2}\left\|\mathbf{v}^{(k+1)}-\mathbf{z}_j\right\|_2^2\} \tag{13}$$

$$\mathbf{p}_j^{(k+1)} = \mathbf{p}_j^{(k)} + \rho(\mathbf{v}^{(k+1)} - \mathbf{z}_j^{(k+1)}) \tag{14}$$

where $\mathbf{p}_j$ is the vector of multipliers of the dual consistency constraints. The superscript in $(.)^{(k+1)}$ is used to indicate the value of a term at the iteration $(k+1)$. It should be noted that since the terms $(\mathbf{p}_j^{(k)})^T\mathbf{z}^{(k)}$ and $(\mathbf{p}_j^{(k)})^T\mathbf{v}^{(k+1)}$ are constant and are simply added to the objectives, they are dropped from Eq. (12) and Eq. (13) respectively.
Eq. (12) is a quadratic optimization problem with respect to $\mathbf{v}$, thus it has an analytical solution:

$$\mathbf{v}^{(k+1)} = \frac{1}{M}\sum_{j=1}^{M}(\mathbf{z}_j) - \frac{1}{M\rho}\sum_{j=1}^{M}\mathbf{p}_j^{(k)} \tag{15}$$

Substituting $d_j(\mathbf{z}_j)$ from Eq. (11) into Eq. (13), we obtain

$$\begin{aligned}\mathbf{z}_j^{(k+1)} &= \arg\max_{\mathbf{z}_j}\{\inf_{\mathbf{y}_j,\mathbf{x}_j,\,\mathbf{g}_j\le\mathbf{0},\mathbf{h}_j=\mathbf{0}}[f_j(\mathbf{y}_j,\mathbf{x}_j) + \mathbf{z}_j{}^T\mathbf{S}_j\mathbf{y}_j] + (\mathbf{p}_j^{(k)})^T\mathbf{z}_j - \frac{\rho}{2}\left\|\mathbf{v}^{(k+1)}-\mathbf{z}_j\right\|_2^2\} \\ &= \arg\max_{\mathbf{z}_j}\{\inf_{\mathbf{y}_j,\mathbf{x}_j,\,\mathbf{g}_j\le\mathbf{0},\mathbf{h}_j=\mathbf{0}}[f_j(\mathbf{y}_j,\mathbf{x}_j) + \mathbf{z}_j{}^T(\mathbf{S}_j\mathbf{y}_j + \mathbf{p}_j^{(k)}) - \frac{\rho}{2}\left\|\mathbf{v}^{(k+1)}-\mathbf{z}_j\right\|_2^2]\}\end{aligned} \tag{16}$$

Let $L(\mathbf{y}_j, \mathbf{x}_j,\mathbf{z}_j)$ represent the objective function in Eq. (16)

$$L(\mathbf{y}_j,\mathbf{x}_j,\mathbf{z}_j) = f_j(\mathbf{y}_j,\mathbf{x}_j) + \mathbf{z}_j{}^T(\mathbf{S}_j\mathbf{y}_j + \mathbf{p}_j^{(k)}) - \frac{\rho}{2}\left\|\mathbf{v}^{(k+1)}-\mathbf{z}_j\right\|_2^2 \tag{17}$$

According to Theorem 37.3 and 37.6 in [18], if $L(\mathbf{y}_j, \mathbf{x}_j, \cdot)$ and $L(\cdot,\cdot, \mathbf{z}_j)$ are convex functions, $L(\mathbf{y}_j, \mathbf{x}_j,\mathbf{z}_j)$ has a saddle point and satisfies

$$\max_{\mathbf{z}_j}\{\min_{\mathbf{y}_j,\mathbf{x}_j,\,\mathbf{g}_j\le\mathbf{0},\mathbf{h}_j=\mathbf{0}}[L(\mathbf{y}_j,\mathbf{x}_j,\mathbf{z}_j)]\} = \min_{\mathbf{y}_j,\mathbf{x}_j,\,\mathbf{g}_j\le\mathbf{0},\mathbf{h}_j=\mathbf{0}}\{\max_{\mathbf{z}_j}[L(\mathbf{y}_j,\mathbf{x}_j,\mathbf{z}_j)]\} \tag{18}$$

Thus, the optimization problem in Eq. (16) becomes

$$\begin{aligned}&\min_{\mathbf{y}_j,\mathbf{x}_j,\,\mathbf{g}_j\le\mathbf{0},\mathbf{h}_j=\mathbf{0}}\{\max_{\mathbf{z}_j}[L(\mathbf{y}_j,\mathbf{x}_j,\mathbf{z}_j)]\} \\ &= \min_{\mathbf{y}_j,\mathbf{x}_j,\,\mathbf{g}_j\le\mathbf{0},\mathbf{h}_j=\mathbf{0}}\{\max_{\mathbf{z}_j}[f_j(\mathbf{y}_j,\mathbf{x}_j) + \mathbf{z}_j{}^T(\mathbf{S}_j\mathbf{y}_j + \mathbf{p}_j^{(k)}) - \frac{\rho}{2}\left\|\mathbf{v}^{(k+1)}-\mathbf{z}_j\right\|_2^2]\}\end{aligned} \tag{19}$$

Since $L(\mathbf{y}_j, \mathbf{x}_j,\mathbf{z}_j)$ is a quadratic function with respect to $\mathbf{z}_j$, the solution to $\max[L(\mathbf{y}_j, \mathbf{x}_j,\mathbf{z}_j)]$ in vector form is

$$\mathbf{z}_j^{(k+1)} = \mathbf{v}^{(k+1)} + \frac{1}{\rho}(\mathbf{p}_j^{(k)} + \mathbf{S}_j\mathbf{y}_j^{(k+1)}) \tag{20}$$

or

$$z_{ij}^{(k+1)} = v_i^{(k+1)} + \frac{1}{\rho}(p_{ij}^{(k)} + (\mathbf{S}_j\mathbf{y}_j^{(k+1)})_i) \tag{21}$$

in scalar form, where $\mathbf{z}_j^{(k+1)} = [z_{1j}^{(k+1)},\ldots,z_{ij}^{(k+1)},\ldots,z_{Nj}^{(k+1)}]^{\mathrm{T}}$.
Substituting Eq. (20) into Eq. (19) we have

$$\begin{aligned}&\min_{\mathbf{y}_j,\mathbf{x}_j,\,\mathbf{g}_j\le\mathbf{0},\mathbf{h}_j=\mathbf{0}}\{f_j(\mathbf{y}_j,\mathbf{x}_j) + [\mathbf{v}^{(k+1)} + \frac{1}{\rho}(\mathbf{p}_j^{(k)} + \mathbf{S}_j\mathbf{y}_j^{(k+1)})]^{\mathrm{T}}(\mathbf{p}_j^{(k)} + \mathbf{S}_j\mathbf{y}_j^{(k+1)}) - \frac{\rho}{2}\left\|\frac{1}{\rho}(\mathbf{p}_j^{(k)} + \mathbf{S}_j\mathbf{y}_j^{(k+1)})\right\|_2^2\} \\ &= \min_{\mathbf{y}_j,\mathbf{x}_j,\,\mathbf{g}_j\le\mathbf{0},\mathbf{h}_j=\mathbf{0}}\{f_j(\mathbf{y}_j,\mathbf{x}_j) + \sum_{i=1}^{N}v_i^{(k+1)}[p_{ij}^{(k)} + (\mathbf{S}_j\mathbf{y}_j^{(k+1)})_i] + \frac{1}{\rho}\sum_{i=1}^{N}[p_{ij}^{(k)} + (\mathbf{S}_j\mathbf{y}_j^{(k+1)})_i]^2 - \frac{1}{2\rho}\sum_{i=1}^{N}[p_{ij}^{(k)} + (\mathbf{S}_j\mathbf{y}_j^{(k+1)})_i]^2\} \\ &= \min_{\mathbf{y}_j,\mathbf{x}_j,\,\mathbf{g}_j\le\mathbf{0},\mathbf{h}_j=\mathbf{0}}\{f_j(\mathbf{y}_j,\mathbf{x}_j) + \frac{1}{2\rho}\sum_{i=1}^{N}[p_{ij}^{(k)} + (\mathbf{S}_j\mathbf{y}_j^{(k+1)})_i]^2 + \sum_{i=1}^{N}v_i^{(k+1)}[p_{ij}^{(k)} + (\mathbf{S}_j\mathbf{y}_j^{(k+1)})_i] + \frac{\rho}{2}\sum_{i=1}^{N}(v_i^{(k+1)})^2 - \frac{\rho}{2}\sum_{i=1}^{N}(v_i^{(k+1)})^2\} \\ &= \min_{\mathbf{y}_j,\mathbf{x}_j,\,\mathbf{g}_j\le\mathbf{0},\mathbf{h}_j=\mathbf{0}}\{f_j(\mathbf{y}_j,\mathbf{x}_j) + \frac{\rho}{2}\sum_{i=1}^{N}[v_i^{(k+1)} + \frac{1}{\rho}(p_{ij}^{(k)} + (\mathbf{S}_j\mathbf{y}_j^{(k+1)})_i)]^2 - \frac{\rho}{2}\sum_{i=1}^{N}(v_i^{(k+1)})^2\}\end{aligned} \tag{22}$$

where ρ and $v_i^{(k+1)}$ are constants and thus the last term in Eq. (**22**) can be dropped. The optimization problem in Eq. (13) is then simplified as

$$\min_{\mathbf{y}_j,\mathbf{x}_j,\,\mathbf{g}_j\le\mathbf{0},\mathbf{h}_j=\mathbf{0}} f_j(\mathbf{y}_j,\mathbf{x}_j) + \frac{\rho}{2}\sum_{i=1}^{N}[v_i^{(k+1)} + \frac{1}{\rho}(p_{ij}^{(k)} + (\mathbf{S}_j\mathbf{y}_j^{(k+1)})_i)]^2 \tag{23}$$

with Eq. (20) as the solution for $\mathbf{z}_j^{(k+1)}$.

2.3. A new parallel coordination algorithm

In summary, the problem in Eq. (6) can be solved through the following steps. The flowchart of this new algorithm is shown in Figure 2.

Step 1: Initialization

Step 2: Compute the master copy of multipliers: $\mathbf{v}^{(k+1)} = \frac{1}{M}\sum_{j=1}^{M}(\mathbf{z}_j) - \frac{1}{M\rho}\sum_{j=1}^{M}\mathbf{p}_j^{(k)}$ (24)

Step 3: For j = 1,...,M, optimize the sub-problem j:

$$\begin{aligned} \underset{\mathbf{y}_j,\mathbf{x}_j}{\text{minimize}} \quad & f_j(\mathbf{y}_j,\mathbf{x}_j) + \frac{\rho}{2}\sum_{i=1}^{P}(v_i^{(k+1)} + \frac{1}{\rho}(p_{ij}^{(k)} + (\mathbf{S}_j\mathbf{y}_j)_i))^2 \\ \text{subject to} \quad & \mathbf{g}_j(\mathbf{y}_j,\mathbf{x}_j) \le \mathbf{0} \\ & \mathbf{h}_j(\mathbf{y}_j,\mathbf{x}_j) = \mathbf{0} \end{aligned} \tag{25}$$

$\mathbf{z}_j^{(k+1)}=(z_{1j}^{(k+1)},...,z_{n_y j}^{(k+1)})$ can be determined by the solution $\mathbf{y}_j^{(k+1)}$:

$$z_{ij}^{(k+1)} = v_i^{(k+1)} + \frac{1}{\rho}(p_{ij}^{(k)} + (\mathbf{S}_j\mathbf{y}_j^{(k+1)})_i) \tag{26}$$

Step 4: check if the optimization converges, if not, compute the Lagrangian multipliers for multipliers and go to step 2

$$\mathbf{p}_j^{(k+1)} = \mathbf{p}_j^{(k)} + \rho(\mathbf{v}^{(k+1)} - \mathbf{z}_j^{(k+1)}) \tag{27}$$

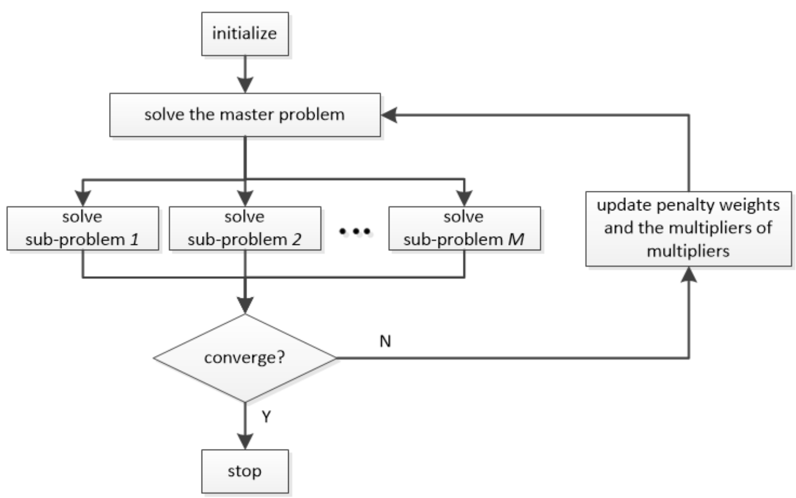


Figure 2: The flowchart of the proposed method

During this process, since copies of shared variables and Lagrangian dual multipliers are generated, consistency constraints are introduced to ensure these copies share the same values. There are two kinds of consistency constraints at iteration k: the primal consistency constraints $\mathbf{c}_p$ and the dual consistency cnstraints $\mathbf{c}_d$.

$$\begin{aligned} \mathbf{c}_p^{(k)} &= \mathbf{S}_1\mathbf{y}_1^{(k)} +,...,+ \mathbf{S}_M\mathbf{y}_M^{(k)} = \mathbf{0} \\ \mathbf{c}_d^{(k)} &= \mathbf{v}^{(k)} - \mathbf{z}_j^{(k)} = \mathbf{0} \end{aligned} \tag{28}$$

$\mathbf{y}_1,\ldots,\mathbf{y}_M$ are shared variables and their copies and $\mathbf{c}_\text{p}$ are used to ensure they have the same values at the end of optimization for the primal problem in Eq. (7). $\mathbf{v}$, $\mathbf{z}_1,\ldots,\mathbf{z}_M$ are dual multipliers and their copies and $\mathbf{c}_\text{d}$ are used to ensure they have the same values at the end of optimization for the dual problem in Eq. (11).

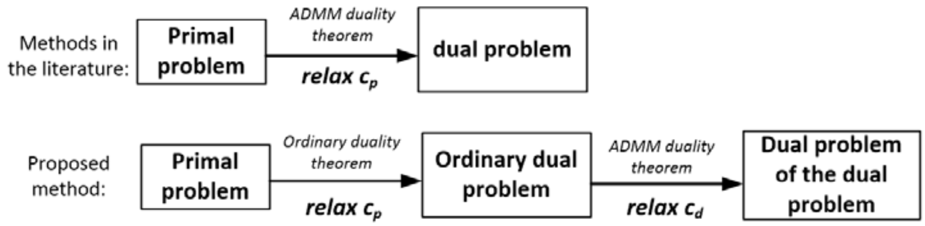


Figure 3：Dual transformations in the proposed method and other methods in the literature

Figure 3 shows how the duality theorems are applied to the methods in literature and the proposed method. Most developed methods apply ADMM directly to the primal problem, relax $\mathbf{c}_p$ into the objective and solve the resulting dual problem instead of solving the original problem. In contrast, in the proposed method two kinds of duality theorems are employed. First the ordinary duality theorem is used to relax $\mathbf{c}_p$ and an ordinary dual problem is generated, to which ADMM is then applied to relax $\mathbf{c}_d$ and a dual problem of the first dual problem is formulated. Instead of solving the original problem, the proposed method is solving this dual problem of the dual problem to achieve the optimal solution for the original problem. Since the ordinary duality is applied initially, the sub-problems in the proposed method are independent from each other and therefore can be solved in parallel. It should be noted that in mathematical programming, the dual problem of the dual problem is supposed to be the primal problem. However, this is not the case here since the two different duality theorems are used in this transformation.

The ultimate goal of the developed algorithm is to solve the primal problem which is difficult to handle directly thus we turn to solving the dual of its dual problem. In view of this, as the convergence criteria of the optimization process only the primal consistency constraints $\mathbf{c}_p$ are checked and $\mathbf{c}_d$ is not used. As long as the primal consistency constraints $\mathbf{c}_p$ satisfy certain conditions and are very close to zero, the current solutions can be considered as consistent and feasible for the primal problem, which is the original problem we want to solve. Thus the convergence criterion for the proposed method are set as

$$\left\|\mathbf{c}_p^{(k)} - \mathbf{c}_p^{(k-1)}\right\|_\infty < \varepsilon \quad \text{and} \quad \left\|\mathbf{c}_p^{(k)}\right\|_\infty < \varepsilon \tag{29}$$

where ε is a convergence tolerance, which normally is set to be very small.

The centralized ALC needs M copies of shared variables while the proposed method only needs M-1. As a result, the number of elements in $\mathbf{c}_p$ in the proposed method is less than that in the centralized ALC. This is expected to bring certain benefits to the proposed method and is explored in the next section.

3. Numerical tests and results analysis

One mathematical problem and one engineering problem are used to test the efficacy of the proposed method. The number of function evaluations is recorded to represent the computational resources consumed. Errors of the design variables and the objective of each problem during the optimization process are also collected. The errors are defined as the relative errors of the decomposition-based optimization results with respect to the reference solution from the All-in-One formulation. All the sub-problem optimizations are solved using *"fmincon"* in Matlab. The centralized ALC-ADMOM is also applied to these test problems for comparison. Through the introduction of an artificial master problem, the centralized ALC-ADMOM allows parallel computation for all decomposed sub-problems and has been proven to be more efficient than other coordination methods [3]. In the following part of this research it is referred to as the centralized ALC for convenience.

3.1. Problem 1 – Geometric optimization problem

The first test problem is a simple geometric optimization problem which has been widely used in the literature [6] [19]. Here the problem is partitioned into two sub-problems and both the centralized ALC and the proposed method need a master problem to coordinate the two sub-problems, as shown in Figure 4. For the centralized ALC, the linking variables between the master problem and the sub-problems are the shared design variables. In the proposed method, the linking variables are the dual multipliers associated with the shared design variables and their copies. In both methods the master problems are simple quadratic programming problems thus their analytical solutions can be easily calculated. The solution process of the master problem is merely a parameter update procedure based on the results of the sub-problems.

$$
\begin{aligned}
\min_{z_1,\ldots,z_7} \quad & f = z_1^2 + z_2^2 \\
\textit{subject to} \quad & g_1 = (z_3^{-2} + z_4^2) z_5^{-2} - 1 \le 0, \\
& g_2 = (z_5^2 + z_6^{-2}) z_7^{-2} - 1 \le 0, \\
& h_1 = (z_3^2 + z_4^{-2} + z_5^2) z_1^{-2} - 1 = 0, \\
& h_2 = (z_5^2 + z_6^2 + z_7^2) z_2^{-2} - 1 = 0,
\end{aligned} \tag{30}
$$

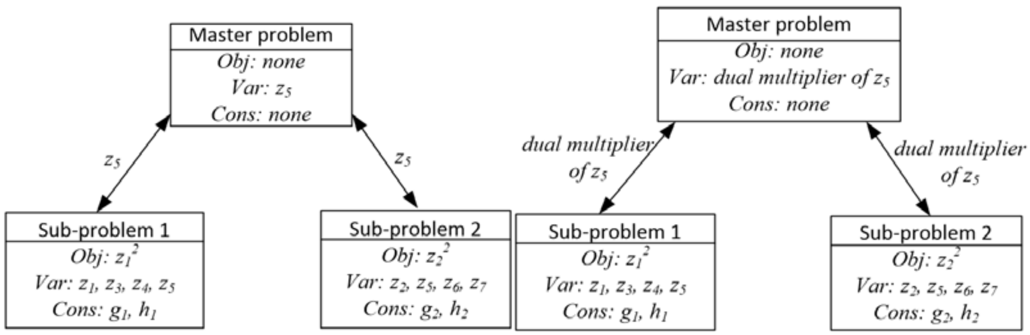
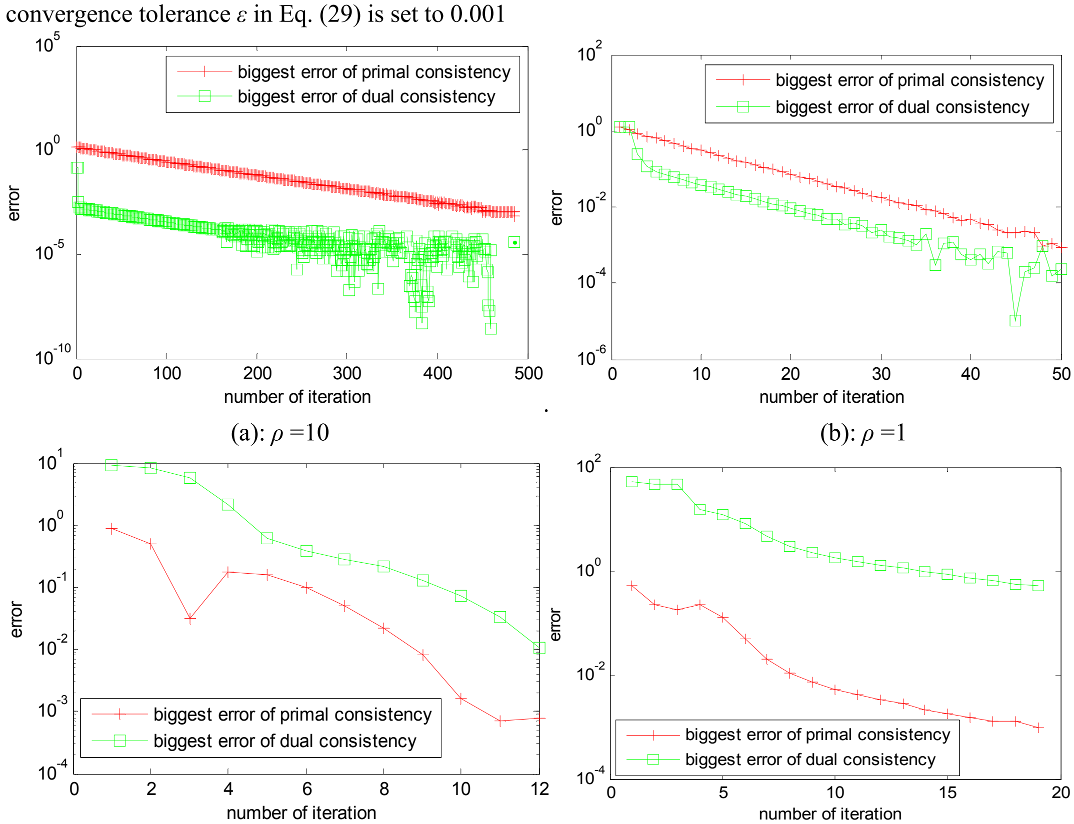


Figure 4：The decompositions of the centralized ALC (left) and the proposed method (right) when solving the geometric optimization problem

To explore the effects of the penalty weight ρ on the primal and dual consistency constraints, different values of ρ, 10, 1, 0.1, and 0.01, are considered. The biggest violations of $\mathbf{c}_p$ and $\mathbf{c}_d$ during the optimization process are plotted in Figure 5 for different ρ. Here the starting points for the design variables in all cases are set to one and the convergence tolerance ε in Eq. (29) is set to 0.001



(a): ρ =10

(b): ρ =1

(c): ρ =0.1

(d): ρ =0.01

Figure 5：The curves of the biggest primal and dual consistency errors under different ρ

Figure 5 clearly indicates that a large ρ can effectively reduce the dual consistency error, which is reasonable because a large ρ places a high penalty on the differences between the copies of dual multipliers. For ρ=10, the dual consistency error reaches around 1e-3 just after a few iteration while it takes the primal consistency error 487 iterations to reach 1e-3. As the value of ρ decreases, the primal consistency error drops faster and the dual consistency drops more slowly. For ρ=0.01, the dual consistency error is around 0.5 when the primal consistency error reaches 1e-3.

Inspired by the above observation, we propose to decrease the value of ρ throughout the optimization process, instead of fixing it to a constant. Eq. (31) is the proposed update scheme for ρ. We want to keep ρ relatively big at the beginning to quickly reduce the dual consistency error, and then we decrease ρ to reduce the primal consistency error faster thus to accelerate the convergence speed of optimization.

$$\rho^{(k+1)} = \beta\rho^{(k)},\ 0<\beta<1 \tag{31}$$

By adjusting the convergence tolerance in Eq. (29), we can control the solution error of decomposition-based optimization. Here, the tolerance ε is set to 1e-2, 1e-3, 1e-4 and 1e-5 for both the centralized ALC and the proposed method. Their performances are collected in Figure 6. The initial ρ in the proposed method and the initial w in centralized ALC are all set to one. The update parameter β for ρ in Eq. (13) is 0.8. For the centralized ALC, we use the weight update scheme in [3] and set β = 1.1 and ϒ = 0.9, as suggested in [3]. The initial Lagrangian multipliers are set to zero, and the initial design variables are set to one. Figure 6 shows the proposed method is more efficient than the centralized ALC in terms of function evaluation numbers.

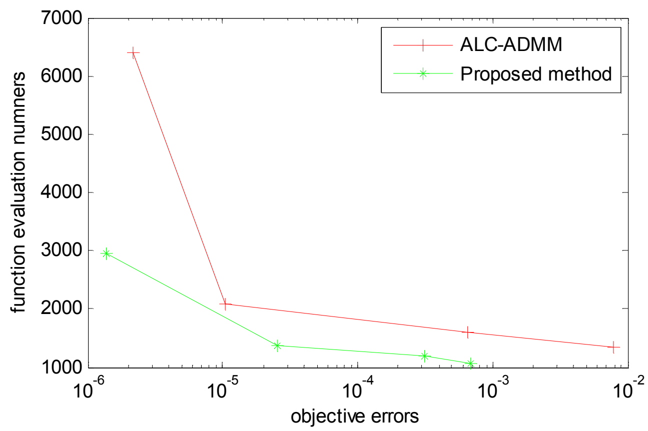


Figure 6: The test results of the centralized ALC and the proposed method on the geometric optimization problem (From right to left: ε = 1e-2, 1e-3, 1e-4 and 1e-5)

3.2. Problem 2 – Portal frame design problem

The second test problem is a structural optimization problem – the portal frame design problem. The optimization objective is to minimize the volume of the whole structure by changing the dimensions of cross sectional areas. Since a horizontal force and a concentrated moment are applied on the structure, normal and shear stresses are incurred. The design constraints are to satisfy all stress limits and geometry requirements. Readers may find more details about this problem in [20].

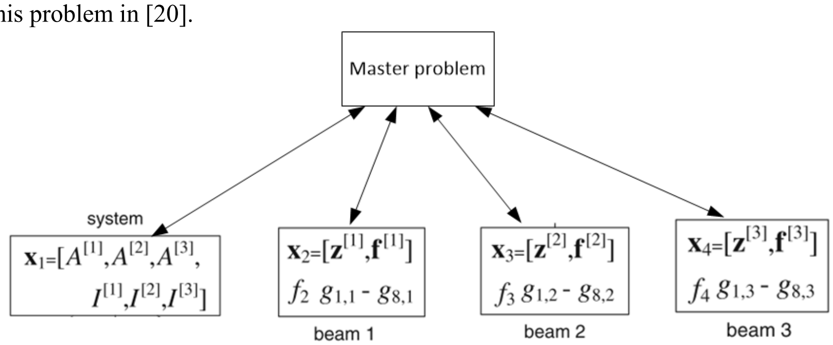


Figure 7: The decompositions of the proposed method and the centralized ALC when solving the portal frame design problem

Due to the uncertainty of one set of the geometry constraints in this problem (which requires that either the top flange area is twice as large as the bottom one or vice versa), this problem has many local optimal solutions and one global optimal solution. Previous research has found that ALC with a hierarchical structure could reach the global solution 54 times out of 100 tries [20]. However the ALC in [20] cannot solve the sub-problems in parallel For this problem, the centralized ALC in [13] is adopted and compared to the proposed method. Both methods use the structure shown in Figure 7.

20 tests are performed for the two methods, starting from random points between the lower bounds and upper bounds of the design variables. The objective errors of these test results are collected in Figure 8. The convergence tolerance ε is set to 1e-3 for both the centralized ALC and the proposed method. For the proposed method, the initial ρ is 1 and β is 0.96. For the centralized ALC, a large initial w such as 1 or 0.1 does not work and always causes the optimization to converge prematurely, so the initial w is set to 0.01 and the update parameters are set as β = 1.05 and ϒ = 0.9. All initial Lagrangian multipliers are set to 0. The AIO optimal objective value is 0.1661, which is used as the global optimal objective value to calibrate the objective error for decomposition-based optimization.

For the centralized ALC, the objective errors for the 20 tests in Figure 8 range from 0.53% to 9.95% with an average value of 3.5%. While for the proposed method, all 20 tests in Figure 8 have reached the same objective value, 0.1660, with the objective error as low as 0.1%, which means the proposed method has greatly improved the solution accuracy and has high robustness. On the efficiency side, the average function evaluation number of the proposed method is 181,421 while that number for the centralized ALC is 368,028.

In a summary, for this multi-modal engineering problem, starting from random design variables, the proposed method consumes only half of the computation resources of the centralized ALC while it reaches a much better solution than the centralized ALC does. Also, the results of the centralized ALC vary a lot as the starting points change while the proposed method can consistently reach an accurate solution.

The numerical tests in this study have proven the high efficiency and accuracy of the proposed method over the centralized ALC. These advantages might from the major fact that the proposed method creates one less set of copies of all shared variables than the centralized ALC, which makes it easier for the algorithm to coordinate and reach a consistent solution. While in the proposed method copies of dual multipliers are introduced, these copies do not necessarily need to be equal to each other for the algorithm to generate the optimal solution for the original problem. For example, the portal frame design problem originally has 24 shared variables. In the proposed method, 24 duplicated shared variables are introduced and thus 24 primal consistency constraints are added to the decomposition formulation. In the centralized ALC, the number of duplicated shared variables is 48 because each original shared variable requires two copies. As a result, the number of primal consistency constraint increases to 48. The convergence criterion for both methods is the closeness of the primal consistency error to zero, and it is apparently much easier to satisfy this condition with 24 rather than 48 primal consistency constraints.

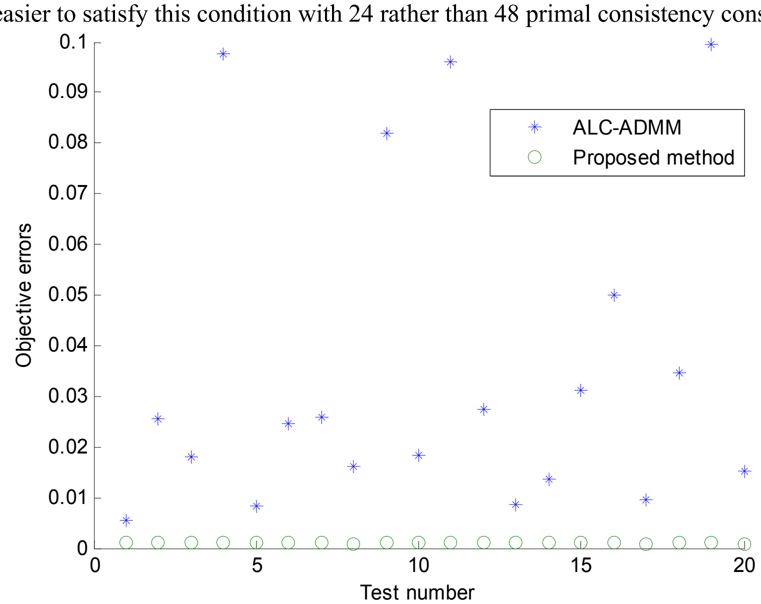


Figure 8: The objective errors of ALC-ADMM and the proposed method for 20 tests of the portal frame problem

4. Conclusions

A new parallel coordination method for decomposition-based optimization is proposed in this paper. Instead of applying ADMM directly to the primal problem, we first employ the ordinary duality theorem to generate an ordinary dual problem, and then apply ADMM to the dual problem. The resulting formulation is the dual problem of the dual problem, in which all sub-problems are independent from each other thus they can be solved in parallel. The mathematical derivations of the method are presented in the paper along with its final computation procedures. One mathematical and one engineering example are used to test the proposed method and the results are compared to those of another popular parallel coordination method: the centralized ALC. The results show that for the mathematical problem, the proposed method consumes less function evaluations than the centralized ALC does while reaching the same objective accuracy. For the complex multi-modal structural optimization problem, the proposed method is more accurate (with an average objective error as 0.1% compared to 3.5% for the centralized ALC), efficient (needs only half of the function evaluation number of the centralized ALC), and robust (consistently reaches the same accurate solution while the solutions of the centralized ALC vary when starting from random initial points).

In the future, more numerical tests need to be conducted to compare the proposed method with other parallel coordination methods. It is also worthwhile to investigate how the penalty weight ρ affects the optimization process, and come up with an improved update scheme for ρ.

5. Acknowledgements

This research was partially supported by the National Science Foundation under grant # CMMI-1129969 entitled "Network Target Coordination for Complex System Optimization". The views presented here do not necessarily reflect those of our sponsor whose support is gratefully acknowledged.

6. References

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

[2] N. Michelena, H. M. Kim, and P. Papalambros, "A System Partitioning and Optimization Approach to Target Cascading", Proceedings of the 12th International Conference on Engineering Design. Vol. 2. 1999.

[3] S. Tosserams, "Distributed Optimization for Systems Design An Augmented Lagrangian Coordination Method," Ph. D. thesis, Eindhoven University of Technology, 2008.

[4] M. Xu, W. Wang, P. Guarneri, and G. Fadel, "CADMM applied to hybrid network decomposition," 10th World Congress on Structural and Multidisciplinary Optimization, Orlando, Florida, May 19-24, 2013.

[5] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein, "Distributed optimization and statistical learning via the alternating direction method of multipliers," Foundations and Trends® in Machine Learning 3, no. 1: 1-122, 2011.

[6] Y. Li, Z. Lu, and J. J. Michalek, "Diagonal Quadratic Approximation for Parallelization of Analytical Target Cascading," Journal of Mechanical Design,130(5), 051402, 2008.

[7] D. P. Bertsekas, Nonlinear Programming, 2nd ed., Athena Scientific, Belmont, MA, 1999.

[8] H. M. Kim, D. G. Rideout, P. Y. Papalambros, and J. L. Stein, "Analytical Target Cascading in Automotive Vehicle Design," Journal of Mechanical Design,125(3), 481-489, 2003.

[9] J. B. Lassiter, M. M. Wiecek, and K. R. Andrighetti, "Lagrangian coordination and Analytical Target Cascading: solving ATC-decomposed problems with Lagrangian duality," Optimization and Engineering, 6(3), 361-381, 2005.

[10] W. Wang, M. Xu, P. Guarneri, G. Fadel, and V. Blouin, "'A Consensus Optimization via Alternating Direction Method of Multipliers for Network Target Coordination.' In 12th AIAA Aviation Technology, Integration, and Operations (ATIO) Conference and 14th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference. 2012.

[11] W. Wang, "Network Target Coordination for Design Optimization of Decomposed Systems," Ph.D. thesis, Clemson University, 2012.

[12] M. Fukushima, "Application of the alternating direction method of multipliers to separable convex programming problems," Computational Optimization and Applications, vol. 1, pp. 93–111, 1992.

[13] S. Tosserams, M. Kokkolaras, L. F. P. Etman, and J. E. Rooda, "A Nonhierarchical Formulation of Analytical Target Cascading," Journal of Mechanical Design, 132(5), p. 051002, 2010.

[14] P. Guarneri, J. T. Leverenz, M. M. Wiecek, and G. Fadel, "Optimization of nonhierarchically decomposed problems," Journal of Computational and Applied Mathematics, vol. 246, pp. 312–319, 2013.

[15] M. Xu, G. Fadel, and M. M. Wiecek, "Solving structure for network-decomposed problems optimizaed with Augmented Lagrangian Coordination," In ASME-IDETC 2014, Buffalo, NY, USA, 2014.

[16] M. Xu, G. Fadel, and M. M. Wiecek, "Dual Residual for Centralized Augmented Lagrangian Coordination Based on Optimality Conditions," Journal of Mechanical Design, 137(6), 2015.

[17] J. J. Michalek and P. Y. Papalambros, "Weights, Norms, and Notation in Analytical Target Cascading," Journal of Mechanical Design, vol. 127, no. 3, p. 499, 2005.

[18] R. T. Rockafellar, Convex Analysis. No. 28. Princeton university press, 1970.

[19] S. Tosserams, L. F. P. Etman, P. Y. Papalambros, and J. E. Rooda, "An augmented Lagrangian relaxation for analytical target cascading using the alternating direction method of multipliers," Structural and Multidisciplinary Optimization, vol. 31, no. 3, pp. 176–189, 2006.

[20] S. Tosserams, L. F. P. Etman, and J. E. Rooda, "Multi-modality in augmented Lagrangian coordination for distributed optimal design," Structural and Multidisciplinary Optimization, vol. 40, no. 1–6, pp. 329–352, 2009.