

DETC2007-35361

A SEQUENTIAL LINEAR PROGRAMMING COORDINATION ALGORITHM FOR ANALYTICAL TARGET CASCADING

Jeongwoo Han*, Panos Papalambros

{jwhan, pyp}@umich.edu

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

ABSTRACT

Decomposition-based strategies, such as analytical target cascading (ATC), are often employed in design optimization of complex systems. Achieving convergence and computational efficiency in the coordination strategy that solves the partitioned problem is a key challenge. A new convergent strategy is proposed for ATC, which coordinates the interactions among sub-problems using sequential linearizations. Linearity of sub-problems is maintained using L_∞ norms to measure deviations between targets and responses. A subproblem suspension strategy is used to temporarily suspend inclusion of subproblems that do not need significant redesign, based on trust region and target value step size. The proposed strategy is intended for use in optimization problems where sequential linearizations are typically effective, such as problems with extensive monotonicities, large number of constraints relative to variables, and propagation of probabilities with normal distributions. Experiments with test problems show that, relative to standard ATC coordination, the number of subproblem evaluations is reduced considerably while maintaining accuracy.

1 Introduction

Optimal system design of complex products often encompasses several interacting subsystems and multidisciplinary anal-

yses. Such problems may be solved with an all-in-one (AIO) method in which the system is treated as a fully integrated single problem. When the analysis models used at each optimization iteration are computationally expensive or difficult to solve, AIO may not be practical or reliable. In such cases, decomposition strategies can be used: the problem is partitioned into several sub-problems that are solved with an appropriate coordination strategy. Decomposition strategies are classified as non-hierarchical or hierarchical. Non-hierarchical partitions often use two levels: sub-problems, typically representing different aspects (or disciplinary analyses), are optimized concurrently, while a system-level problem coordinates the interactions between the sub-problems [1–7].

Analytical target cascading (ATC) is an optimization method for multilevel hierarchical systems typically partitioned into physical subsystems or objects (see Figure 1) [8]. Each block in the hierarchical structure is referred to as an element and is an optimization sub-problem. An element can be coupled with only one parent element but with multiple children elements. The interactions among elements with the same parents, the so-called siblings, are not linked directly to each other but are coordinated by their parent. The linking variables between a parent and children are design targets and analysis responses. Targets are set by parents and propagated to their children; the children solve a minimum deviation optimization problem to obtain responses that are as close to the targets as possible. Thus, targets and re-

*Corresponding author, Phone/Fax: (734) 647-8402/8403

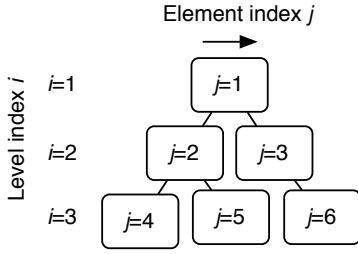


Figure 1. Example of index notation for a hierarchically partitioned design problem [9]

sponses are updated and coordinated iteratively to achieve consistent values within all elements where they appear.

ATC enforces consistency of values shared between elements by using penalty functions. The proper choice of penalty functions and associated weights is critical for solution convergence. For quadratic penalty functions, large weights are required to obtain accurate and consistent solutions [10]. Similar to other decomposition strategies, ATC typically is more expensive than AIO, if the latter could be used to obtain a solution, due to the coordination overhead. Michalek et al. developed an iterative method for updating weights, which finds minimal weights to achieve a given level of inconsistency, especially important for problems with unattainable system targets [9]. Still, the inner loop coordination, where the decomposed ATC problems are solved iteratively, is computationally expensive. To address this, Tosserams et al. [11] introduced a separable augmented Lagrangian penalty function and an alternating direction solution method, resulting to significant reduction in computational cost. The original ATC problems were deterministic but probabilistic ATC (PATC) problems have also been recently formulated and solved [12, 13]. In such problems propagation of uncertainty in nonlinear optimization problems makes computations very expensive. If normal distributions are assumed, such propagation will be easy to compute if the problems were linear. There are other situations where algorithms based on linearizations of nonlinear problems can be advantageous, for example the presence of monotonicity and large number of inequality constraints [14, 15].

In this article, we employ sequential linear programming (SLP) as an alternate coordination strategy to solve ATC problems: the elements in the hierarchy are linearized and the linearized ATC is solved successively. The inspiration for the particular algorithm comes from recent SLP-filter implementations on probabilistic optimization problems [16, 17]. In the proposed algorithm, the linearized subproblems have significantly lower levels of complexity and can be solved easily. Solving the linearized ATC requires no system analysis function evaluation during the inner loop coordination, and so the associated cost is rel-

atively small, especially for problems with expensive analyses. Also, the analyses in the decomposed elements can be executed concurrently. In addition to the SLP algorithm, a suspension strategy, similar to that in [18], is used to avoid analyses of elements that do not need substantial redesign, for example, when a child element has a weaker coupling to its parent than those of the other children.

The article is organized as follows. In Section 2, SLP-based ATC is formulated and the notational modifications on SLP-filter algorithm are explained. Section 3 proposes the suspension strategy for SLP-based ATC and explains the algorithm flow. Illustrative test examples are presented in Section 4, followed by conclusions in Section 5.

2 SLP-based ATC

In SLP-based ATC, a nonlinear ATC problem is linearly approximated and solved using the “standard” ATC strategy to obtain the optimal solution of the Linearized ATC (LATC) problem. By solving LATC problems successively, the algorithm converges to a solution of the original nonlinear problem with the aid of a filter algorithm and trust region method [19]. To maintain linearity, the LATC formulation requires different penalty functions than those used in other ATC formulations [11, 20]. Decomposition and relaxation errors lead to a modification of the SLP-filter algorithms developed by Fletcher et al. [19]. The details of LATC formulation are explained in Section 2.1 while Section 2.2 discusses the proposed modifications.

2.1 Linearized ATC formulation

We consider an AIO system design problem similar to that in Tosserams [11], expressed as

$$\begin{aligned} \min_{\mathbf{x}} f(\mathbf{x}) \\ \text{subject to } \mathbf{g}(\mathbf{x}) \leq \mathbf{0}, \\ \mathbf{h}(\mathbf{x}) = \mathbf{0}, \end{aligned} \quad (1)$$

where \mathbf{x} is the vector of all design variables, f is the system objective function, and \mathbf{g} and \mathbf{h} are inequality and equality constraint functions, respectively. Even though the convergence proof of the SLP algorithm in [19] is presented with only inequality constraints, the filter algorithm can be extended for problems with equality constraints using a constraint violation function similar to that defined in [21].

Assume that the AIO problem in Eq.(1) can be decomposed hierarchically into M elements of N levels. Then the quantities with indices ij are related to element j at level i (See Figure 1). Local design variables of element j at level i are indicated as \mathbf{x}_{ij} , whose couplings with its parent are represented through

target variables \mathbf{t}_{ij} , similar to [11], and response copies \mathbf{r}_{ij} are introduced to make the constraint set fully separable. By assuming that $f(\mathbf{x})$ is also additively separable through the response copies and introducing consistency constraints \mathbf{c}_{ij} , the modified AIO problem is presented as:

$$\begin{aligned}
& \min_{\bar{\mathbf{x}}_{11}, \dots, \bar{\mathbf{x}}_{NM}} \sum_{i=1}^N \sum_{j \in \mathcal{E}_i} f_{ij}(\bar{\mathbf{x}}_{ij}) \\
& \text{subject to } \mathbf{g}_{ij}(\bar{\mathbf{x}}_{ij}) \leq \mathbf{0}, \\
& \quad \mathbf{h}_{ij}(\bar{\mathbf{x}}_{ij}) = \mathbf{0}, \\
& \quad \mathbf{c}_{ij} = \mathbf{t}_{ij} - \mathbf{r}_{ij} = \mathbf{0}, \\
& \text{where } \bar{\mathbf{x}}_{ij} = [\mathbf{x}_{ij}, \mathbf{r}_{ij}, \mathbf{t}_{(i+1)k}], \quad \forall k \in C_{ij} \\
& \quad \forall j \in \mathcal{E}_i, i = 1, \dots, N.
\end{aligned} \tag{2}$$

In Eq.(2), f_{ij} , \mathbf{g}_{ij} and \mathbf{h}_{ij} are the separated objective, inequality and equality constraints of element j at level i , respectively; C_{ij} is the set of the children of element j at level i , and \mathcal{E}_i is the set of elements at level i . Note that the solution from Eq.(2) solves the original nonlinear problem Eq.(1).

For the SLP convergence argument presented in Section 2.2, a linear approximation is applied before decomposition. Note that applying the decomposition first will result in the same final LATC formulation. The LP problem of the modified AIO depends on the value of $\bar{\mathbf{x}}_{ij}^{(l)}$ ($\forall j \in \mathcal{E}_i, i = 1, \dots, N$) at an SLP iteration l and trust region radius $\rho^{(l)}$ ($\rho^{(l)} > 0$), and is given by:

$$\begin{aligned}
& \min_{\bar{\mathbf{d}}_{11}^{(l)}, \dots, \bar{\mathbf{d}}_{NM}^{(l)}} \sum_{i=1}^N \sum_{j \in \mathcal{E}_i} \nabla f_{ij}^T(\bar{\mathbf{x}}_{ij}^{(l)}) \bar{\mathbf{d}}_{ij}^{(l)} \\
& \text{subject to } \nabla \mathbf{g}_{ij}^T(\bar{\mathbf{x}}_{ij}^{(l)}) \bar{\mathbf{d}}_{ij}^{(l)} + \mathbf{g}_{ij}(\bar{\mathbf{x}}_{ij}^{(l)}) \leq \mathbf{0}, \\
& \quad \nabla \mathbf{h}_{ij}^T(\bar{\mathbf{x}}_{ij}^{(l)}) \bar{\mathbf{d}}_{ij}^{(l)} + \mathbf{h}_{ij}(\bar{\mathbf{x}}_{ij}^{(l)}) = \mathbf{0}, \\
& \quad \mathbf{c}_{ij}^{(l)} = \mathbf{t}_{ij}^{(l)} + \mathbf{d}_{\mathbf{t}_{ij}}^{(l)} - \mathbf{r}_{ij}^{(l)} - \mathbf{d}_{\mathbf{r}_{ij}}^{(l)} = \mathbf{0}, \\
& \quad \|\bar{\mathbf{d}}_{ij}^{(l)}\|_{\infty} \leq \rho^{(l)}, \\
& \text{where } \bar{\mathbf{d}}_{ij}^{(l)} = [\mathbf{d}_{\mathbf{x}_{ij}}^{(l)}, \mathbf{d}_{\mathbf{r}_{ij}}^{(l)}, \mathbf{d}_{\mathbf{t}_{(i+1)k}}^{(l)}], \quad \forall k \in C_{ij} \\
& \quad \forall j \in \mathcal{E}_i, i = 1, \dots, N.
\end{aligned} \tag{3}$$

In Eq.(3), the L_{∞} norm is used to define the trust region because its implementation requires only simple bounds to the LAIO problem.

Allowing inconsistencies amongst elements or relaxing the consistency constraints \mathbf{c}_{ij} enable decomposition. Therefore, the overall system can be consistent at convergence by minimizing the deviation between elements, ε_{ij} , throughout the ATC iterations. Previous ATC formulations utilize three types of relaxations that are added to the objective: quadratic penalty [8,9,20],

ordinary Lagrangian [22] and augmented Lagrangian relaxations [11]. In this article, a weighted L_{∞} norm is applied in order to maintain the linearity of the elements:

$$\|\mathbf{w}_{ij} \circ (\mathbf{t}_{ij} - \mathbf{r}_{ij})\|_{\infty} \leq \varepsilon_{ij} \Rightarrow \max |\mathbf{w}_{ij} \circ (\mathbf{t}_{ij} - \mathbf{r}_{ij})| \leq \varepsilon_{ij} \tag{4}$$

where the \circ operation indicates the component-wise multiplication of two vectors such that $\{a_1, \dots, a_k\}^T \circ \{b_1, \dots, b_k\}^T = \{a_1 b_1, \dots, a_k b_k\}^T$. The outcome of $\mathbf{w}_{ij} \circ (\mathbf{t}_{ij} - \mathbf{r}_{ij})$ is a $1 \times m_{ij}$ vector, where m_{ij} is the number of components in \mathbf{t}_{ij} or \mathbf{r}_{ij} . The right-hand-side equation is reformulated into a minimization problem with $2m_{ij}$ constraints:

$$\begin{aligned}
& \min \varepsilon_{ij} \\
& \text{subject to } -\tilde{\varepsilon}_{ij} \leq \mathbf{w}_{ij} \circ (\mathbf{t}_{ij} - \mathbf{r}_{ij}) \leq \tilde{\varepsilon}_{ij},
\end{aligned} \tag{5}$$

where $\tilde{\varepsilon}_{ij}$ is a $1 \times m_{ij}$ vector of ε_{ij} . By combining Eq.(5) with Eq.(3) as a relaxation term, the relaxed LAIO problem is given by:

$$\begin{aligned}
& \min \sum_{i=1}^N \sum_{j \in \mathcal{E}_i} \nabla f_{ij}(\bar{\mathbf{x}}_{ij}^{(l)}) \bar{\mathbf{d}}_{ij}^{(l)} + \sum_{i=1}^N \sum_{j \in \mathcal{E}_i} \varepsilon_{ij}^{(l)} \\
& \text{find } \bar{\mathbf{d}}_{11}^{(l)}, \dots, \bar{\mathbf{d}}_{NM}^{(l)}, \varepsilon_{22}^{(l)}, \dots, \varepsilon_{NM}^{(l)} \\
& \text{subject to } \nabla \mathbf{g}_{ij}^T(\bar{\mathbf{x}}_{ij}^{(l)}) \bar{\mathbf{d}}_{ij}^{(l)} + \mathbf{g}_{ij}(\bar{\mathbf{x}}_{ij}^{(l)}) \leq \mathbf{0}, \\
& \quad \nabla \mathbf{h}_{ij}^T(\bar{\mathbf{x}}_{ij}^{(l)}) \bar{\mathbf{d}}_{ij}^{(l)} + \mathbf{h}_{ij}(\bar{\mathbf{x}}_{ij}^{(l)}) = \mathbf{0}, \\
& \quad -\tilde{\varepsilon}_{ij}^{(l)} \leq (\mathbf{w}_{ij} \circ (\mathbf{t}_{ij} + \mathbf{d}_{\mathbf{t}_{ij}} - \mathbf{r}_{ij} - \mathbf{d}_{\mathbf{r}_{ij}}))^{(l)} \leq \tilde{\varepsilon}_{ij}^{(l)}, \\
& \quad \|\bar{\mathbf{d}}_{ij}^{(l)}\|_{\infty} \leq \rho^{(l)}, \\
& \text{where } \bar{\mathbf{d}}_{ij}^{(l)} = [\mathbf{d}_{\mathbf{x}_{ij}}^{(l)}, \mathbf{d}_{\mathbf{r}_{ij}}^{(l)}, \mathbf{d}_{\mathbf{t}_{(i+1)k}}^{(l)}], \quad \forall k \in C_{ij} \\
& \quad \forall j \in \mathcal{E}_i, i = 1, \dots, N,
\end{aligned} \tag{6}$$

Note that, unlike quadratic or augmented Lagrangian penalty functions, most of the consistency constraints will remain inactive unless ε_{ij} becomes zero. Therefore, we cannot use monotonicity analysis [14] to incorporate the consistency constraints into the objective function.

While the convergence property of ATC has been proven based on quadratic penalty functions [20] and a modified Lagrangian dual formulation [23], the proof with a L_{∞} norm has yet to be provided. To ensure a similar level of consistency in convergence as in these other methods, slightly larger weights are expected at the optimum because the L_{∞} norm used in this study underestimates the deviation compared to a L_2 norm. It is important to mention that a L_2 norm will provide the same result with proper selection of weights at the system consistency ($\varepsilon_{ij} = 0$).

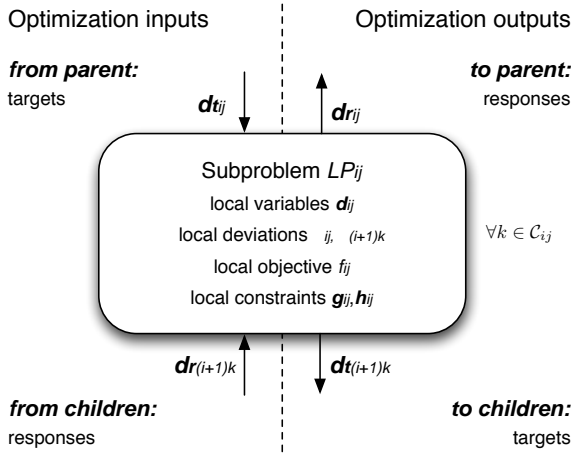


Figure 2. Information flow for analytical target cascading (ATC) subproblem LP_{ij} of Eq.(7) (modified from [11])

By decomposing the problem into separable elements, the LATC subproblem LP_{ij} of element j at level i is formulated as

$$\begin{aligned}
& \min \nabla f_{ij}^T(\bar{\mathbf{x}}_{ij}) \bar{\mathbf{d}}_{ij} + \varepsilon_{ij} + \sum_{k \in C_{ij}} \varepsilon_{(i+1)k} \\
& \text{find } \bar{\mathbf{d}}_{ij}, \varepsilon_{ij}, \varepsilon_{(i+1)1}, \dots, \varepsilon_{(i+1)n_{ij}} \\
& \text{subject to } \nabla \mathbf{g}_{ij}^T(\bar{\mathbf{x}}_{ij}) \bar{\mathbf{d}}_{ij} + \mathbf{g}_{ij}(\bar{\mathbf{x}}_{ij}) \leq \mathbf{0}, \\
& \quad \nabla \mathbf{h}_{ij}^T(\bar{\mathbf{x}}_{ij}) \bar{\mathbf{d}}_{ij} + \mathbf{h}_{ij}(\bar{\mathbf{x}}_{ij}) = \mathbf{0}, \\
& \quad -\tilde{\varepsilon}_{ij} \leq (\mathbf{t}_{ij} + \mathbf{d}_{t_{ij}} - \mathbf{r}_{ij} - \mathbf{d}_{r_{ij}}) \leq \tilde{\varepsilon}_{ij}, \\
& \quad -\tilde{\varepsilon}_{(i+1)k} \leq \{\mathbf{w} \circ (\mathbf{t} + \mathbf{d}_{\mathbf{t}} - \mathbf{r} - \mathbf{d}_{\mathbf{r}})\}_{(i+1)k} \leq \tilde{\varepsilon}_{(i+1)k}, \\
& \quad \|\bar{\mathbf{d}}_{ij}\|_{\infty} \leq \rho, \\
& \text{where } \bar{\mathbf{d}}_{ij} = [\mathbf{d}_{\mathbf{x}_{ij}}, \mathbf{d}_{\mathbf{r}_{ij}}, \mathbf{d}_{\mathbf{t}_{(i+1)k}}], \quad \forall k \in C_{ij},
\end{aligned} \tag{7}$$

where iteration index l is dropped for convenience. Information flows to and from a subproblem LP_{ij} are presented in Figure 2.

With a proper selection of weights in ATC, deviation errors become zero at convergence [20]. An important observation in Eq.(7) is that the global and local convergence properties depend on the size of trust regions and the scaling of design variables. In other words, the size of the $\bar{\mathbf{d}}_{ij}$ may be limited by the linking variables if the scaling of the design variables is not appropriate, which may cause the SLP algorithm to converge more slowly than an AIO formulation.

2.2 Notational modifications of SLP-filter algorithm

In the SLP-filter algorithm presented by Fletcher et al. [19], every LP solution is evaluated for the system objective and con-

straints in order to check that the solution is acceptable to the current filter and the linear approximation is proper. Since the system objective and constraints are separated into elements, an equivalent system objective f_e is required, and defined by:

$$f_e(\bar{\mathbf{x}}_{11}, \dots, \bar{\mathbf{x}}_{NM}) = f(\mathbf{z}) = \sum_{i=1}^N \sum_{j \in \mathcal{E}_i} f_{ij}(\bar{\mathbf{x}}_{ij}). \tag{8}$$

The equivalent predicted and actual reduction in $f_e(\bar{\mathbf{x}}_{11}, \dots, \bar{\mathbf{x}}_{NM})$ are denoted as Δl_e and Δf_e , respectively, and calculated as

$$\begin{aligned}
\Delta l_e &= \nabla f_e^T(\bar{\mathbf{x}}_{11}, \dots, \bar{\mathbf{x}}_{NM}) \{\bar{\mathbf{d}}_{\bar{\mathbf{x}}_{11}}, \dots, \bar{\mathbf{d}}_{\bar{\mathbf{x}}_{NM}}\} \\
&= \sum_{i=1}^N \sum_{j \in \mathcal{E}_i} \nabla f_{ij}^T \bar{\mathbf{d}}_{ij}
\end{aligned} \tag{9}$$

$$\begin{aligned}
\Delta f_e &= f_e(\bar{\mathbf{x}}_{11}, \dots, \bar{\mathbf{x}}_{NM}) - f_e(\bar{\mathbf{x}}_{11} + \bar{\mathbf{d}}_{11}, \dots, \bar{\mathbf{x}}_{NM} + \bar{\mathbf{d}}_{NM}) \\
&= \sum_{i=1}^N \sum_{j \in \mathcal{E}_i} f_{ij}(\bar{\mathbf{x}}_{ij}) - \sum_{i=1}^N \sum_{j \in \mathcal{E}_i} f_{ij}(\bar{\mathbf{x}}_{ij} + \bar{\mathbf{d}}_{ij})
\end{aligned} \tag{10}$$

The terms in Eq.(9) and Eq.(10), including ∇f_{ij}^T , $f_{ij}(\bar{\mathbf{x}}_{ij})$ and $f_{ij}(\bar{\mathbf{x}}_{ij} + \bar{\mathbf{d}}_{ij})$, are easily and independently obtained from the decomposed elements. Note that the deviation errors in Eq.(7) are not included in either the predicted or the actual reduction calculation. Instead, they are treated as additional equality constraints, and constraint violation functions η are expressed similarly to those in [21, 24],

$$\eta(\bar{\mathbf{x}}_{11}, \dots, \bar{\mathbf{x}}_{NM}) = \|\mathbf{g}_{ij}^+(\bar{\mathbf{x}}_{ij})\|_1 + \|\{\mathbf{h}_{ij}(\bar{\mathbf{x}}_{ij}), \mathbf{c}_{ij}(\bar{\mathbf{x}}_{ij})\}\|_1. \tag{11}$$

The term η replaces h in the publications cited above to avoid notational confusion; \mathbf{g}_{ij}^+ is a vector of constraint violation functions, $g^+ = \max(0, g)$. The decomposed constraints, $\mathbf{g}_{ij}^+(\bar{\mathbf{x}}_{ij})$, $\mathbf{h}_{ij}(\bar{\mathbf{x}}_{ij})$ and $\mathbf{c}_{ij}(\bar{\mathbf{x}}_{ij})$, can be obtained independently. Based on the values in Eq.(8 -11), acceptability to the current filter \mathcal{F} is determined similarly to [19].

$$\eta \leq \beta \eta_{i_{\mathcal{F}}} \quad \text{or} \quad f_e \leq (f_e)_{i_{\mathcal{F}}} - \gamma \eta_{i_{\mathcal{F}}}, \quad \forall i_{\mathcal{F}} \in \mathcal{F} \tag{12}$$

A trial point $\{\eta, f_e\}$, acceptable to \mathcal{F} , is regarded as an f-type iteration (improving f_e with a possible increase in η), an η -type iteration (reducing h with a possible increase in f_e), or an unacceptable point.

$$\text{f-type iteration} : \Delta f_e^{(l)} \geq \sigma \Delta l_e^{(l)} \quad \text{and} \quad \Delta l_e^{(l)} \geq \delta (\eta^{(l)})^2 \tag{13}$$

$$\text{\eta-type iteration} : \Delta l_e^{(l)} < \delta (\eta^{(l)})^2 \tag{14}$$

$$\text{not acceptable} : \text{otherwise.} \tag{15}$$

Note that the top level target in Eq.(7) may not be attainable in the early iterations due to small trust regions. For problems with unattainable targets, relaxation in the LATC formulation will result in arbitrarily small inconsistency deviations, if weights are chosen appropriately [9]. Additionally, filters ensure that η converges to zero as SLP iterations continue. Thus, the system inconsistency is minimized twice through LATC and SLP-filter algorithms and the solution from Eq.(7) converges to the solution obtained from the LAIO problem, Eq.(3).

While the convergence of LATC with a L_∞ norm is not proven directly, it can be readily proven that the sequence of solutions of Eq.(3) converges to the solution of Eq.(2). Since the solution from Eq.(2) equals the solution from Eq.(1), the equivalence of solutions from Eq.(2) and Eq.(3) needs to be proven. Under standard assumptions (the non-empty design space is well-bounded and the objective and constraint functions are twice continuously differentiable) the SLP-filter algorithm is proven to converge to a Karush-Kuhn-Tucker (KKT) point or an accumulation point that satisfies a Fritz-John condition for problems without equality constraints [19]. The algorithm can be readily extended to problems with equality constraints under a Mangasarian-Fromowitz constraint qualification (MFCQ), an extended form of the Fritz-John condition in the presence of equality and inequality constraints [25, 26].

If the standard assumptions are satisfied and the original problem is compatible within a round-off error, the SLP filter algorithm (A) finds a KKT point or (B) has an infinite subsequence of consecutive f-type or η -type iterations [19]. Now we will show that if (B) occurs, the algorithm converges to a feasible point and, if MFCQ holds, the set of directions \mathbf{s} in Eq.(16) is empty:

$$\{\mathbf{s} \mid \mathbf{s}^T \nabla f_e < 0, \quad \mathbf{s}^T \nabla \mathbf{g}_A < \mathbf{0}, \quad \mathbf{s}^T \nabla \mathbf{h} = \mathbf{0}, \quad \mathbf{s}^T \nabla \mathbf{c} = \mathbf{0}\}, (16)$$

where \mathbf{g}_A is the active inequality constraints.

The trust region radius $\rho^{(l)}$ decreases and ultimately $\rho^{(l)} \rightarrow 0$. When the trust region is reduced, a trial point will be found that is acceptable, and either an f-type iteration or an η -type iteration will occur. For the resulting $\rho^{(l)}$,

$$\begin{cases} \text{if } \rho^{(l)} \leq (1 - \sigma)\varepsilon/M, \text{ then } \Delta f_e^{(l)} > \sigma \Delta l_e^{(l)}, \\ \text{if } (\rho^{(l)})^2 \leq \beta \tau^{(l)}/mM, \text{ then } \eta(\mathbf{x}^{(l)} + \mathbf{d}^{(l)}) \leq \beta \tau^{(l)}, \end{cases} (17)$$

where $0 < \varepsilon \leq \min\{-(\nabla f_e^{(l)})^T \mathbf{s}^{(l)}, -(\nabla \mathbf{g}_A^{(l)})^T \mathbf{s}^{(l)}\}$, $\tau^{(l)} = \min_{i \in \mathcal{F}} \eta_i$, m is the number of all constraints, and M is the upper bound for $\|\frac{1}{2} \mathbf{d}^T (\nabla^2 f_e) \mathbf{d}\|$, $\|\frac{1}{2} \mathbf{d}^T (\nabla^2 \mathbf{g}) \mathbf{d}\|_\infty$, $\|\frac{1}{2} \mathbf{d}^T (\nabla^2 \mathbf{h}) \mathbf{d}\|_\infty$ and $\|\frac{1}{2} \mathbf{d}^T (\nabla^2 \mathbf{c}) \mathbf{d}\|_\infty$. Also, $\mathbf{s}^{(l)}$ is the unit vector of the projection of \mathbf{d} to the space spanning the equality constraints.

Let the sequence of $\mathbf{x}^{(l)}$ of (B) converge to \mathbf{x}^∞ . From the assumption that the objective function f_e is bounded, the filter envelop test in Eq.(12) ensures $\sum \eta^{(l+1)}$ is bounded and $\eta^{(l)} \rightarrow 0$, so \mathbf{x}^∞ is feasible [19]. The main sequence contains an infinite number of f-type or η -type iterations. Then we assume that MFCQ is satisfied and consider the proposition (to be contradicted) that \mathbf{x}^∞ is not a KKT point. Then, it is always possible to find a solution of Eq.(3) with a trust region satisfying Eq.(18) from [19, 21].

$$\frac{\eta^{(l)}}{\varepsilon} \leq \rho \leq \min \left\{ \frac{(1 - \sigma)\varepsilon}{M}, \frac{\varepsilon}{M}, \frac{\bar{c}^{(l)}}{\bar{a}^{(l)}}, \frac{\sigma\varepsilon}{\gamma mM}, \sqrt{\frac{\beta \tau^{(l)}}{mM}} \right\} (18)$$

where $0 \leq \bar{c}^{(l)} \leq -\max\{\mathbf{g}_A^{(l)}\}$ and $\bar{a}^{(l)} \geq \max\{(\nabla \mathbf{g}_A^{(l)})^T \mathbf{s}\}$. \mathbf{g}_A denotes the vector of inactive inequality constraints. For sufficiently large l , it is not possible for any value of $\eta^{(l)} \leq \varepsilon \rho$ to satisfy Eq.(14) since $\Delta l_e^{(l)}$ decreases monotonically as ρ decreases. Thus, for sufficiently large l , f-type iterations are always generated. Then both left- and right-sides of Eq.(18) remain constant because $\eta^{(l)}$ and $\tau^{(l)}$ are not updated. Therefore, for sufficiently large l , Δf_e does not converge to zero. This contradicts the fact that f_e is bounded. Thus by contradiction, for a sufficiently large l , the algorithm has an accumulation point that is feasible and is either a KKT point or fails to satisfy MFCQ.

If the original nonlinear problem of Eq.(1) is assumed to be well-bounded and have a solution, the sequence of solutions of Eq.(3) should converge to the solution of Eq. (1) if targets and responses are bounded. That is, if the decomposed systems are coupled through variables that are well-bounded, the solution of Eq.(3) converges to the point that satisfies the necessary conditions for solving Eq.(1).

3 Suspension strategy for SLP-based ATC

Even though LATC and LAIO converge to a very similar solution with small relaxation errors, LATC may require more computations to obtain a converged solution than LAIO due to coordination costs. During the LATC solution, however, the objectives and constraints are not evaluated. Indeed, in this study we focus on the reduction in the number of function evaluations during the SLP-based ATC algorithm, rather than improvement of LATC performance, such as the convergence rate or deviation errors in the ATC strategy. To this end, we take advantage of decomposed design tasks.

Linking variables represent couplings between elements. Elements that are weakly coupled will likely be less sensitive to linking variables than to local variables. Therefore, changes in the target for an element that are sufficiently small will not have significant impact on the system objective. In this case, the change in the target can be neglected during a given iteration and the element in the branch can be ‘‘suspended’’ from redesign (or

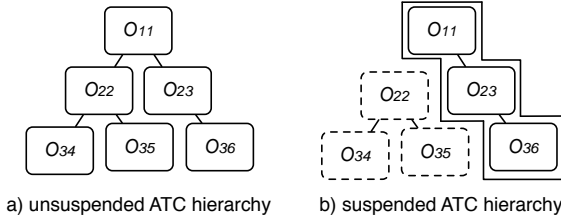


Figure 3. Examples of the unsuspended and suspended ATC hierarchies

evaluation) [18, 27, 28]. For example, let us assume that the step size of t_{22} in Figure 3 is considerably smaller than that of t_{23} at iteration k . Then the elements in the corresponding branch, including O_{22} , O_{34} and O_{35} , can be suspended from evaluation for $f_{ij}^{(k+1)}$, $g_{ij}^{(k+1)}$ and $h_{ij}^{(k+1)}$ and the values at iteration k are used instead while the actual functions in the unsuspended element (e.g. O_{11} , O_{23} and O_{36} in Figure 3) are evaluated to obtain Δf_e and η .

The ATC with suspension strategy flowchart is shown in Figure 4. The steps in the suspension strategy (solid boxes) are described in more detail below.

Step A: Attempt to solve $LATC(\bar{\mathbf{x}}_{ij}^{(l)}, \rho^{(l)})$

Eq. 7 is solved using a “standard” ATC strategy to obtain $\bar{\mathbf{d}}_{ij}^{(l)}$. If any elements in the system are incompatible, a feasibility restoration phase is evoked in order to find a point that is both acceptable and compatible. Once the LATC strategy converges to a solution that is compatible, the step sizes are compared in order to determine active and suspended elements in the next step.

Step B: Apply suspension criteria

This step selects the set of elements that can be suspended based on the step size of the solution’s targets from the previous step. With the solution of element j at level i , the step sizes for targets $\mathbf{d}_{(i+1)k}^{(l)}$ are compared. If a target to a child m satisfies

$$\|\mathbf{d}_{(i+1)m}^{(l)}\| < \zeta_t \sum_{k \in C_{ij}} \frac{\|\mathbf{d}_{(i+1)k}^{(l)}\|}{NC_{ij}}, \quad (19)$$

then the elements in the corresponding branch are selected for suspension. The parameter $\zeta_t \ll 1$ is chosen based on the designer’s experience and NC_{ij} is the number of children of element j at level i . The suspension idea is similar to that in [18], and so the modified global sensitivity equations (MGSEs) may provide an estimation for the impact of the target changes that is accurate enough to reduce iteration between steps B and C. In this

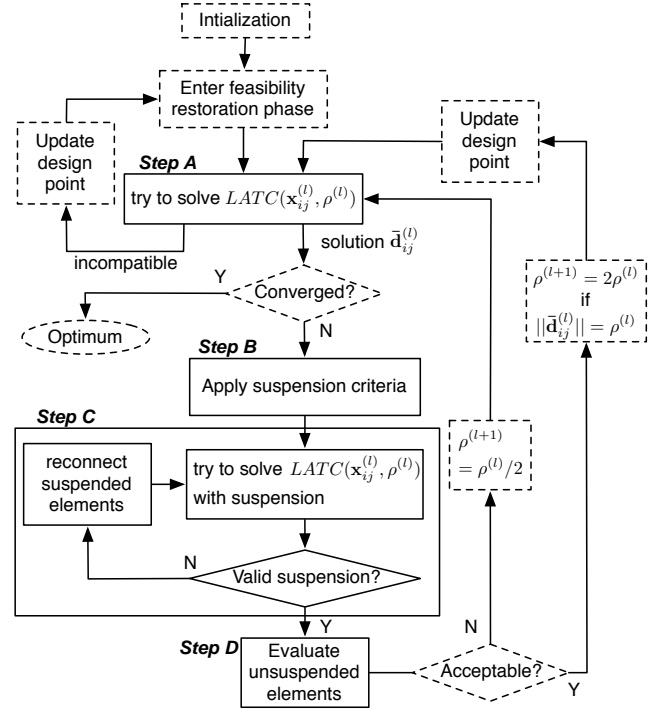


Figure 4. Flowcharts of SLP-based ATC with suspension strategy

study, however, simple comparison on step sizes is used because MGSEs may require additional function evaluations.

The predicted reduction without suspension Δl_e needs to be obtained from the solution for validation at Step C.

Step C: Suspension validation

In this step, LATC is solved again after suspending the elements selected in the previous step to estimate the effect of suspended elements. The suspension can be readily implemented by setting the responses of the suspended elements to zero ($\mathbf{d}_{(i+1)m}^{(l)} = 0$). Once the solution is obtained, the predicted reduction in f_e with suspended elements Δl_e^{sus} is calculated. If $\Delta l_e^{\text{sus}} \geq \zeta_f \Delta l_e$, the suspension is assumed to be valid. The parameter $\zeta_f < 1$ is also chosen based on the designers experience. Otherwise, the suspension is declared to be inadequate and some of the suspended elements must be reactivated if more than two elements are selected for suspension. After reactivation, LATC is solved again until suspension is valid or all elements are active.

Step D: Evaluation

Only active elements are evaluated because the step sizes of suspended elements are all set to zero in the previous step. For the suspended elements, the values from iteration k can be used.

The suspension criteria presented here, including the validation step, can be made conservative by setting $\zeta_t \rightarrow 0$, $\zeta_f \rightarrow 1$. The numerical examples presented in Section 4 show that the number of function evaluations saved by the suspension strategy depends highly on the values of ζ_t and ζ_f . Also the convergence of the suspension strategy is not guaranteed. However, the method remains attractive in design problems because even if convergence is not attained, the intermediate solutions are feasible and usually represent an improvement in the objective function. Defining parameters ranges that guarantee both convergence and reduction in function evaluations is a subject for further research.

4 Numerical results

This section provides two test examples to illustrate the proposed algorithm. Both examples have three elements: one at the top level and two at the bottom level. One child element is more weakly coupled to the top level element than the other since the proposed suspension strategy is expected to be more effective on problems whose elements have significantly different coupling strengths.

In the examples, ATC problems are formulated from AIO problems and solved with the SLP-filter algorithm. Results from calculations with and without the suspension strategy are compared to each other, and also to results from the original AIO problems solved by SLP-filter and SQP algorithms.

4.1 Example 1: Modified Hock and Schittkowski Problem 34

Problem 34 in the classical test collection by Hock and Schittkowski [29] is modified so that the problem can be decomposed into three elements. The original AIO problem is

$$\begin{aligned} \min_{x_1, \dots, x_6} f &= -x_1 x_4 \\ \text{subject to } g_1 &\equiv \exp(x_1) - x_2 x_5 \leq 0, \\ g_2 &\equiv \exp(x_2) - x_3 \leq 0, \\ g_3 &\equiv \log(5x_4^2) - x_5 \leq 0, \\ g_4 &\equiv x_5^2 - 10x_6 \leq 0, \end{aligned} \quad (20)$$

where the lower and upper bounds of \mathbf{x} are $\{0, 0, 0, 0.01, 0, 0\}$ and $\{100, 100, 10, 100, 100, 5\}$. The unique optimal solution is $\mathbf{x}^* = \{2.79, 2.30, 10.00, 15.35, 7.07, 5.00\}$ with all constraints active.

Eq.(20) is decomposed into one top-level element (O_{11}) with two children (O_{22} and O_{23}). The linking variables that couple O_{11} with O_{22} and O_{23} are x_2 and x_5 , respectively. Then O_{11} minimizes the sum of system objective f and deviation errors ϵ_{22}

Table 1. Optimal solutions and number of redesigns for Example 1

	\mathbf{x}^*	number of redesigns			
		O_{11}	O_{22}	O_{23}	total
<i>LATC</i>	{2.79, 2.30, 10.00, 15.35, 7.07, 5.00}	56 [†]	36 [†]	36 [†]	128 [†]
<i>LATC-SS</i>	{2.79, 2.30, 10.00, 15.35, 7.07, 5.00}	56 [†]	36 [†]	17 [†]	109 [†]
<i>ATC</i>	{2.79, 2.30, 10.00, 15.35, 7.07, 5.00}	777 [†]	164 [†]	154 [†]	1095 [†]
<i>AIO-SLP</i>	{2.79, 2.30, 10.00, 15.35, 7.07, 5.00}				98 [‡]
<i>AIO-SQP</i>	{2.79, 2.30, 10.00, 15.35, 7.07, 5.00}				112 [‡]

[†] Element O_{11} contains one objective and two constraints while element O_{22} and O_{23} include one constraint function.

[‡] AIO problem contains one objective and four constraints.

and ϵ_{23} with respect to $\bar{\mathbf{x}}_{11} = \{x_1, x_4, t_{x_2}, t_{x_5}, \epsilon_{22}, \epsilon_{23}\}$, subject to g_1 and g_3 . O_{22} minimizes ϵ_{22} with respect to $\bar{\mathbf{x}}_{22} = \{x_3, r_{x_2}, \epsilon_{22}\}$, subject to g_2 , while O_{23} minimizes ϵ_{23} with respect to $\bar{\mathbf{x}}_{23} = \{x_6, r_{x_5}, \epsilon_{23}\}$, subject to g_4 . Parameters for the suspension strategy are set to $\zeta_t = 0.2$ and $\zeta_f = 0.8$, and the initial trust region is set to 20.

Table 1 summarizes the final solutions and the number of function evaluations for each element obtained from the five algorithms: *LATC* denotes the results obtained from the SLP-based ATC strategy without suspension strategy, *LATC-SS* denotes the SLP-based ATC strategy with suspension strategy, *ATC* denotes the “standard” ATC strategy with a quadratic penalty function, *AIO-SLP* denotes the SLP-filter algorithm solving the AIO problem and *AIO-SQP* denotes the SQP algorithm solving the AIO problem. All algorithms converge to the same solution. The L_∞ norms of consistency errors for *LATC*, *LATC-SS* and *ATC* are 3.55×10^{-15} , 1.78×10^{-15} and 4.57×10^{-10} , respectively.

Note that element O_{11} requires three functions to be calculated (f , g_1 , g_3), while O_{22} and O_{23} require one function (g_2 and g_4). If we define the computation cost of *LATC* to be $(3 \times 56 + 36 + 36)$ and use it as a baseline, then the normalized computational costs of *LATC-SS*, *ATC*, *AIO-SLP* and *AIO-SQP* are 0.92, 11.0, 2.04 and 2.33, respectively. Here we assume that the computational cost of function evaluation is larger than that of the coordination overhead.

The problem is monotonic for all design variables, and so SLP algorithms have fast convergence. In addition to the effect of monotonicity, decomposition reduces significantly the number of computations by eliminating unnecessary gradient calculations. The suspension strategy also reduces computations by reducing the number of redesigns for O_{23} . Figure 5 shows the history of target values for O_{22} and O_{23} . The step sizes of x_5 are sufficiently smaller than those of x_2 at iterations 2, 3, 5, 7 and 9, and so O_{23} is suspended at these iterations and the previous values of g_4 and ∇g_4 are used instead.

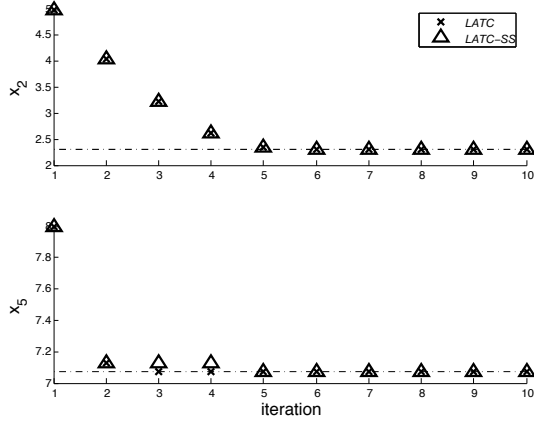


Figure 5. History of targets to element O_{22} and element O_{23}

4.2 Example 2: Allison's structural optimization problem

The second example is a structural optimization problem based on the analytical mass allocation problem of Allison et al. [30] and Tosserams et al. [11] with some modifications: In the hierarchy, the element at the second level (the middle bar) is relocated to the top level and the other two are located at the bottom level. The coupling strength between the second and third rod is strengthened. The original AIO problem is

$$\begin{aligned}
& \min_{d_1, d_2, d_3, d_{r1}, d_{r2}} \sum_{i=1}^3 m_i + \sum_{j=1}^2 m_{r,j} \\
& \text{subject to } g_{1,i} \equiv \sigma_{b,i} - \bar{\sigma} \leq 0 \quad i = 1, 2, 3 \\
& \quad g_{2,j} \equiv \sigma_{a,j} - \bar{\sigma} \leq 0 \quad j = 1, 2 \\
& \quad g_{3,i} \equiv (F_i - F_{i+1}) - \bar{F}_{t,i} \leq 0 \quad i = 1, 2, 3 \\
& \quad g_4 = f_1 - \bar{f}_1 \leq 0 \\
& \quad h_{1,j} \equiv f_j - f_{j+1} - f_{r,j} = 0 \quad j = 1, 2 \\
& \text{where } m_i = \frac{\pi}{4} d_i^2 L \rho, \quad \sigma_{b,i} = \frac{32L(F_i - F_{i+1})}{\pi d_i^3}, \\
& \quad f_i = \frac{64L^3(F_i - F_{i+1})}{3\pi E_i d_i^4}, \quad i = 1, 2, 3; \\
& \quad m_{r,j} = \frac{\pi}{4} d_{r,j}^2 L \rho, \quad \sigma_{a,j} = \frac{4F_{j+1}}{\pi d_{r,j}^2}, \\
& \quad f_{r,j} = \frac{4F_{j+1}L}{\pi E_{r,j} d_{r,j}^2} \quad j = 1, 2
\end{aligned} \tag{21}$$

where m_i is the mass of beam i , $m_{r,j}$ is the mass of rod j , $\sigma_{b,i}$ is the bending stress in beam i , $\sigma_{a,j}$ is the axial stress in rod j , f_i is the vertical deflection of beam i and $f_{r,j}$ is the elongation of rod j . Constraint limits for stress ($\bar{\sigma}$), transmitted force (\bar{F}_t) and vertical deflection of beam 1 (f_1) are set to $127 \cdot 10^6 \text{N/m}^2$, 400N and 27mm , respectively. The equality constraints can be solved explicitly to obtain F_2 and F_3 . The length of beams

Table 2. Optimal solutions and the number of redesigns for Example 2

	\mathbf{x}^*	number of redesigns			
		O_{11}	O_{22}	O_{23}	total
<i>LATC</i>	{34.62, 34.84, 25.22, 40.11, 37.52}	445 [†]	329 [†]	271 [†]	1045 [†]
<i>LATC-SS</i>	{34.62, 34.84, 25.22, 40.11, 37.52}	486 [†]	127 [†]	288 [†]	901 [†]
<i>ATC</i>	{34.62, 34.84, 25.22, 40.11, 37.52}	895 [†]	336 [†]	506 [†]	1737 [†]
<i>AIO-SLP</i>	{34.62, 34.84, 25.22, 40.11, 37.52}				547 [‡]
<i>AIO-SQP</i>	{34.62, 34.84, 25.22, 40.11, 37.52}				418 [‡]

[†] Elements O_{11} , O_{22} and O_{23} contain five, six and three functions, respectively.

[‡] AIO problem contains one objective and eleven constraints.

and rods L and the density of the material ρ are fixed to be 1m and 2700kg/m^3 , respectively. Similar to [11], 1000N is vertically applied at the end of beam 1 ($F_1 = 1000 \text{N}$). In order to apply different coupling strengths, the Young's moduli of the beams and rods are set differently, such as $E_1 = E_2 = E_{r,1} = 70 \text{GPa}$, $E_3 = 700 \text{GPa}$, $E_{r,2} = 7 \text{GPa}$. Therefore, the coupling strength between beams 2 and 3 becomes significantly stronger than that between beam 1 and 2. The lower and upper bounds of \mathbf{x} are set to $\{0.001, 0.001, 0.001, 0.0001, 0.0001\}$ and $\{0.06, 0.06, 0.06, 0.006, 0.006\}$.

The local variables at the top-level element O_{11} are the dimensions of beam 2 and rod 2 ($\mathbf{x}_{11} = \{d_2, d_{r,2}\}$) while those at the two bottom-level elements, O_{22} and O_{23} , are the dimensions of beam 1 and rod 1, and beam 3, respectively ($\mathbf{x}_{22} = \{d_1, d_{r,1}\}$ and $\mathbf{x}_{23} = \{d_3\}$). Element O_{11} is coupled with O_{22} and O_{23} through corresponding axial forces and deflections that are $\{F_2, f_2\}$ and $\{F_3, f_3\}$, respectively. Then O_{11} minimizes the sum of $f_{11} = m_2 + m_{r,2}$ and deviation errors ϵ_{22} and ϵ_{23} , subject to $g_{1,2}$, $g_{2,2}$, $g_{3,2}$ and $h_{1,2}$. O_{22} minimizes the sum of $f_{22} = m_1 + m_{r,1}$ and ϵ_{22} , subject to $g_{1,1}$, $g_{2,1}$, $g_{3,1}$, g_4 and $h_{1,1}$, while O_{23} minimizes the sum of $f_{23} = m_3$ and ϵ_{23} , subject to $g_{1,3}$ and $g_{3,3}$. Since SLP-based algorithms are more effective when problems are well-scaled, the diameters of beams and rods are multiplied by 1000 and 10000, respectively. The scaled initial point is $\mathbf{x}_s^0 = \{30, 30, 25, 30, 30\}$. Parameters for the suspension strategy are set to $\zeta_r = 0.2$ and $\zeta_f = 0.8$, and the initial trust region is set to 0.4.

Table 2 summarizes the final solutions and the number of function evaluations for each element obtained from the five algorithms. The results show that the proposed algorithms converge to the same solution obtained from the other algorithms solving the AIO problem. The L_∞ norms of consistency errors of *LATC*, *LATC-SS* and *ATC* are 6.21×10^{-4} , 1.65×10^{-4} and 2.98×10^{-3} , respectively. Unlike the previous example, the initial point and initial size of trust regions affect significantly the convergence of the SLP-based algorithms. The three elements include different numbers of functions to be evaluated, and the

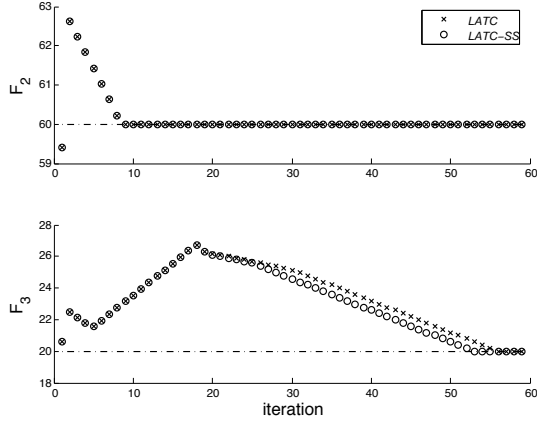


Figure 6. History of targets to element O_{22} and element O_{23}

normalized computational cost of *LATC-SS*, *ATC*, *AIO-SLP* and *AIO-SQP* are 0.81, 1.60, 1.31 and 1.00, respectively.

Without linearity and monotonicity, SLP-based algorithms show no advantages over a sequential quadratic programming algorithm. On the other hand, applying the suspension strategy halves the number of redesigns in O_{22} even though the number of redesigns in the other elements increases slightly. Figure 6 shows the history of target values for O_{22} and O_{23} . Since O_{22} reaches the optimum within 10 iterations, this element is evaluated only a few times during the remaining iterations until O_{23} converges to the optimum.

Tradeoffs between computational cost reductions in O_{22} and increases in O_{11} and O_{23} are observed by varying ζ_t and ζ_f . Since the selection of ζ_t and ζ_f is based on user experience, a more rigorous measure that is less sensitive to problem types needs to be developed.

5 Conclusion

SLP-filter algorithms were introduced into an ATC formulation to reduce computational costs for some problem classes. The L_∞ norm is employed to maintain linearity of the consistency constraints. Since some of \mathbf{c}_{ij} cannot be active unless strict consistency is satisfied, deviation errors ϵ_{ij} remain in the objective functions. Even if \mathbf{c}_{ij} may cause computational inefficiency due to degeneracy when \mathbf{w}_{ij} and $\mathbf{t}_{ij} - \mathbf{r}_{ij}$ are small, numerical results show that the effect is not substantial during the inner loop coordination with sufficiently large \mathbf{w}_{ij} .

Some notation used in previous SLP-filter algorithm formulations was modified here so that definitions are equivalent to those in [19]. For convergence of the proposed LATC algorithm, both convergence proofs of SLP-filter algorithms (Eq.(1) and Eq.(3)) and ATC (Eq.(3) and Eq.(7)) need to hold. For the first part of the proof, the convergence proof of SLP-filter algorithm in [19] was extended to problems with equality constraints

and holds for the decomposed problems. But the second step of the proof (ATC convergence) is not rigorously proven because the effect of the L_∞ norm on ATC convergence has yet to be understood. The examples in Section 4 show that the proposed SLP-based ATC converges to the solution accurately.

Decomposition enables suspension strategy to be used in order to reduce the number of function evaluation taking advantage of the properties of weakly-couple elements. Even though the suspension criteria do not guarantee either reduced computational cost or global convergence, numerical experiments presented in Section 4 show 10 to 20% reduction in computational cost with the proper selection of parameters based on normalized computational costs depending on the balance of coupling strengths. Results must be compared further with other ATC relaxation methods that have shown better numerical efficiency and convergence, such as augmented Lagrangian functions [11].

The suspension strategy can be applied to other decomposition methods, such as collaborative optimization. Suspended elements are recognized as objects that do not need significant design changes in a hierarchical decomposition. Suspended elements in a non-hierarchical decomposition could be aspects or disciplines insensitive to system design changes. Promising results from the examples give a limited demonstration of the SLP-filter algorithms advantages. They warrant further investigation of the method applied to more complex design problems, including probabilistic optimization problems the original inspiration for development of this method.

ACKNOWLEDGMENT

This work was partially supported by the General Motors Corporation, the Automotive Research Center, a US Army Center of Excellence in Modeling and Simulation of Ground Vehicle Systems, at the University of Michigan, and NSF Grant DMI-0503737. This support is gratefully acknowledged.

REFERENCES

- [1] Alexandrov, N. M., and Lewis, R. M., 2002. "Analytical and computational aspects of collaborative optimization for multidisciplinary design". *AIAA Journal*, **40**(2), pp. 301 – 309.
- [2] Balling, R., and Sobieszcanski-Sobieski, J., 1996. "Optimization of coupled systems: A critical overview of approaches". *AIAA Journal*, **34**(1), pp. 6 – 17.
- [3] Braun, R., Kroo, I., and Moore, A., 1996. "Use of the collaborative optimization architecture for launch vehicle design". In NASA, and ISSMO, Symposium on Multidisciplinary Analysis and Optimization, no. AIAA-1996-4018.
- [4] Cramer, E. J., J. E. Dennis, J., Frank, P. D., Lewis, R. M., and Shubin, G. R., 1994. "Problem formulation for multidisciplinary optimization". *SIAM Journal on Optimization*, **4**(4), pp. 754–776.
- [5] Kroo, I., Altus, S., Braun, R., Gage, P., and Sobieski, I., 1994. "Multidisciplinary optimization methods for aircraft preliminary

- design”. In AIAA/USAF/NASA/ISSMO Symposium on Multidisciplinary Analysis and Optimization, no. AIAA 1994-4325, pp. 697–707.
- [6] Kroo, I., 1997. “Multidisciplinary optimization applications in preliminary design - status and directions”. In AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference and Exhibit, 38th, and AIAA/ASME/AHS Adaptive Structures Forum, Kissimmee, FL, Apr. 7-10, no. AIAA-1997-1408.
- [7] Tappeta, R., and Renaud, J., 1997. “Multiobjective collaborative optimization”. *Journal of Mechanical Design, Transactions of the ASME*, **119**(3), pp. 403 – 411.
- [8] Kim, H. M., Michelena, N., Papalambros, P., and Jiang, T., 2003. “Target cascading in optimal system design”. *Trans. ASME, J. Mech. Des.*, **125**(3), pp. 474 – 80.
- [9] Michalek, J., and Papalambros, P., 2005. “An efficient weighting update method to achieve acceptable consistency deviation in analytical target cascading”. *Trans. ASME, J. Mech. Des.*, **127**(2), pp. 206 – 14.
- [10] Bertsekas, D. P., 1999. *Nonlinear programming*, 2nd ed. Belmont, Mass. : Athena Scientific.
- [11] Tosserams, S., Etman, L., Papalambros, P., and Rooda, J., 2006. “An augmented lagrangian relaxation for analytical target cascading using the alternating direction method of multipliers”. *Struct. Multidiscip. Optim.*, **31**(3), pp. 176 – 89.
- [12] Kokkolaras, M., Mourelatos, Z. P., and Papalambros, P. Y., 2006. “Design optimization of hierarchically decomposed multilevel systems under uncertainty”. *Journal of Mechanical Design, Transactions of the ASME*, **128**(2), pp. 503 – 508.
- [13] Liu, H., Chen, W., Kokkolaras, M., Papalambros, P., and Kim, H., 2006. “Probabilistic analytical target cascading: a moment matching formulation for multilevel optimization under uncertainty”. *Trans. ASME, J. Mech. Des.*, **128**(4), pp. 991 – 1000.
- [14] Papalambros, P., and Wilde, D., 2000. *Principles of Optimal Design: Modeling and Computation*, 2nd ed. Cambridge University Press, Cambridge.
- [15] Loh, H. T., and Papalambros, P., 1991. “A sequential linearization approach for solving mixed-discrete nonlinear design optimization problems”. *Trans. ASME, J. Mech. Des.*, **113**(3), pp. 325 – 34.
- [16] Chan, K.-Y., Skerlos, S. J., and Papalambros, P., 2007. “An adaptive sequential linear programming algorithm for optimal design problems with probabilistic constraints”. *Journal of Mechanical Design*, **129**(2), pp. 140–149.
- [17] Chan, K.-Y., Skerlos, S., and Papalambros, P. Y., 2006. “Monotonicity and active set strategies in probabilistic design optimization”. *Journal of Mechanical Design*, **128**(4), pp. 893–900.
- [18] Alyaout, S. F., Papalambros, P. Y., and Ulsoy, A. G., 2005. “Quantification and use of system coupling in decomposed design optimization problems”. *American Society of Mechanical Engineers, Computers and Information in Engineering Division, CED*, **10**, pp. 95 – 103.
- [19] Fletcher, R., Leyer, S., and Toint, P., 1998. “On the global convergence of an SLP-filter algorithm”. *Numerical Analysis Report NA/183, University of Dundee, UK*, **98**(13), pp. 1–11.
- [20] Michelena, N., Park, H., and Papalambros, P. Y., 2003. “Convergence properties of analytical target cascading”. *AIAA Journal*, **41**(5), pp. 897 – 905.
- [21] Fletcher, R., Leyffer, S., and Toint, P. L., 2002. “On the global convergence of a filter–SQP algorithm”. *SIAM Journal on Optimization*, **13**(1), pp. 44–59.
- [22] Lassiter, J. B., Wiecek, M. M., and Andrighetti, K. R., 2005. “Lagrangian coordination and analytical target cascading: Solving atc-decomposed problems with lagrangian duality”. *Optimization and Engineering*, **6**(3), pp. 361–381.
- [23] Kim, H. M., Chen, W., and Wiecek, M. M., 2006. “Lagrangian coordination for enhancing the convergence of analytical target cascading”. *AIAA Journal*, **44**(10), pp. 2197 – 2207.
- [24] Fletcher, R., and Leyffer, S., 2002. “Nonlinear programming without a penalty function”. *Mathematical Programming*, **V91**(2), pp. 239–269.
- [25] Bazaraa, M. S., Sherali, H. D., and Shetty, C. M., 2006. *Nonlinear Programming: Theory and Algorithms*, 3rd ed. John Wiley & Sons.
- [26] Mangasarian, O. L., 1969. *Nonlinear Programming*. McGraw-Hill, New York.
- [27] English, K., Bloebaum, C., and Miller, E., 2001. “Development of multiple cycle coupling suspension in the optimization of complex systems”. *Structural and Multidisciplinary Optimization*, **22**(4), pp. 268 – 283.
- [28] Kasarekar, N. T., and English, K. W., 2004. “Development of a hybrid MDF/IDF multidisciplinary optimization solution method with coupling suspension”. *Collection of Technical Papers - 10th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference*, **3**, pp. 1865 – 1874.
- [29] Hock, W., and Schittkowski, K., 1981. *Test Examples for Nonlinear Programming Codes*. Springer-Verlag New York, Inc., Secaucus, NJ, USA.
- [30] Allison, J., Kokkolaras, M., Zawislak, M., and Papalambros, P., 2005. “On the use of analytical target cascading and collaborative optimization for complex system design”. In Proceedings of the 6th World Congress on Structural and Multidisciplinary Optimization.