AIAA JOURNAL
Vol. 51, No. 3, March 2013

Technical Notes

TECHNICAL NOTES are short manuscripts describing new developments or important results of a preliminary nature. These Notes should not exceed 2500 words (where a figure or table counts as 200 words). Following informal review by the Editors, they may be published within a few months of the date of receipt. Style requirements are the same as for regular contributions (see inside back cover).

Application of Collaborative Optimization to Optimal Control Problems

Xiang Li*
Beijing Institute of Technology, 100081 Beijing, People's Republic of China
ChangAn Liu†
Xi'an Aerospace Propulsion Institute, 710100 Xi'an, Shaanxi Province, People's Republic of China
Weiji Li‡
Northwestern Polytechnical University, 710100 Xi'an, Shaanxi Province, People's Republic of China
and
Haibin Shang**
Beijing Institute of Technology, 100081 Beijing, People's Republic of China

DOI: 10.2514/1.J052006

I. Introduction

FROM the view point of practical computation, multidisciplinary design optimization (MDO) can be considered as methods for solving complex optimization problems. We think that MDO is, in some sense, a bridge between the conventional optimization algorithms and complex applications. There are two main strategies used in these MDO methods, which are approximation and decomposition. Although these two strategies are not mutually exclusive, they are applicable to problems with different properties, respectively.

Some reported flight-vehicle configuration shape-optimization-design problems, integrated with complex analysis models (e.g., computational fluid dynamics or computational structural mechanics), have a small number of design variables and constraints [1,2]. Methods using approximation, such as surrogate-based methods, are applicable to these problems in which the original complex analysis models are replaced by corresponding approximate and relatively simple models, such as radial basis function and Kriging models. The optimization computation is then performed based on these approximate models.

Received 4 April 2012; revision received 21 August 2012; accepted for publication 22 August 2012; published online 28 November 2012.

*Associate Professor, School of Aerospace Engineering; lixiang_0504@yahoo.com.
†Senior Engineer, Xi'an Aerospace Propulsion Institute.
‡Professor, School of Aeronautics.
**Associate Professor, School of Aerospace Engineering.

On the other hand, a flight-vehicle trajectory optimization-design problem with constraints of differential equations, also called optimal control, can be viewed as an infinite-dimensional extension of a common nonlinear optimization problem [3], which is a practical solution that is to convert the infinite-dimensional problem into a finite-dimensional problem. Several conversion methods, for example, direct shooting, multiple direct shooting, collocation, and pseudospectral, have been developed [3,4]. In some cases, the conversion will result in a very high-dimensional nonlinear optimization problem with a large number of design variables and constraints [5]. Some optimizers for large-scale optimizations, such as SNOPT [6], have been presented. In this Note, an alternative solution using collaborative optimization (CO), an MDO method with decomposition strategy, is discussed. Compared with approximation, decomposition is more applicable to this kind of large-scale problem in which the original large-scale problem is decomposed into several reduced subproblems. That is to say, the original complex computation task of one optimization is decomposed into several relatively small computation tasks of several optimizations. Although the computational difficulties in CO have not been solved ideally, the decomposition strategy is a natural and potential way for solving optimization problems with a large number of design variables and constraints. As far as we know, the solution to this kind of large-scale optimization problem converted from optimal control problems by using decomposed method has not been reported, which is the main motivation of the work in this Note.

The organization of this Note is as follows: in Sec. II, we briefly review the collocation method for converting an optimal control problem into a nonlinear optimization problem, and then the discussions on the decomposition of the converted problem using CO are presented in Sec. III. In Sec. IV, a numerical test case illustrates the discussions in Secs. II and III. The conclusions are stated in Sec. V.

II. Problem Formulation

Problems covered in this Note are defined as follows:

$$\text{Min } J = \phi(Y(t_f), X(t_f), P) \tag{1a}$$

$$\text{s.t. } C_s:\ \frac{\mathrm{d}Y(t)}{\mathrm{d}t} - F(Y(t), X(t), P) = 0 \tag{1b}$$

$$C_b:\ C_b(Y(t_0), X(t_0), Y(t_f), X(t_f), P) = 0 \tag{1c}$$

$$C_t:\ C_t(Y(t), X(t), P) \le 0 \tag{1d}$$

where $X(t) = [x_1(t) \ldots x_m(t)]$ is a vector of m-dimensional unknown dynamic variables (also called control variables), e.g., the law of angle of attack varying with time. $P = [p_1 \ldots p_q]$ is a q-dimensional vector of unknown static variables, e.g., the lunch azimuth. $Y(t) = [y_1(t) \ldots y_r(t)]$ is a vector of r-dimensional state variables governed by state differential equations [Eq. (1b)], e.g., the altitude varying with time. $F = [f_1 \ldots f_r]$ describes the dynamic relations of these variables, e.g., the flight vehicle six degrees of freedom (DOF) dynamic equations. From the viewpoint of MDO, $Y(t)$ can also be treated as unknown dynamic design variables, and Eq. (1b) is then treated as r-dimensional equality constraints, $C_s = [c_{s1} \ldots c_{sr}]$. $C_b = [c_{b1} \ldots c_{bg}]$ is a vector of g-dimensional equality constraints of boundary conditions. $C_t = [c_{t1} \ldots c_{th}]$ is a vector of h-dimensional inequality constraints. t_0 and t_f are the starting time and final time, respectively. In some cases, t_0 and t_f are unknown and should also be treated as static design variables. The formulation of problem (1) is similar to that of a general MDO problem [7], whereas an obvious difference lies in the state equations, that is, the state equations in [7] are algebraic equations and these in problem (1) are differential equations.

In this study, a collocation method is used for converting problem (1) into a finite-dimensional nonlinear optimization problem; the key steps of which are described in the following part. Detailed discussions on the collocation method can be found in [3,4].

The time interval $[t_0 \ldots t_f]$ is discretized into n subintervals $[t_0, t_1], \ldots, [t_{n-1}, t_n]$ $(t_n = t_f)$; the length of the ith segment is h^i. $X(t_i)$ and $Y(t_i)$ $(i = 0, \ldots, n)$ at each time node t_i are then treated as the design variables, for convenience, $X(t_i)$ and $Y(t_i)$ are written as X^i and Y^i in the remainder of the Note, respectively. Using Hermite interpolation cubic polynomials are defined for each state variable on each segment. Then problem (1) is converted into a finite-dimensional nonlinear optimization problem as follows:

$$\text{Find } [Y^0 \ldots Y^n, X^0 \ldots X^n, P] \quad \text{Min } J = \phi(Y^n, X^n, P) \tag{2a}$$

$$\text{s.t. } C_s^i\colon\ Y^i - Y^{i+1} + \frac{h^i}{6}(F^i + 4F^{ic} + F^{i+1}) = 0 \quad i = 0, 1, \ldots, n-1 \tag{2b}$$

$$C_t^i\colon\ C_t^i(Y^i, X^i, P) \le 0 \qquad i = 0, 1, \ldots, n \tag{2c}$$

$$C_b\colon\ C_b(Y^0, X^0, Y^n, X^n, P) = 0 \tag{2d}$$

where $F^{ic} = F(Y^{ic}, X^{ic}, P)$, superscript "^{ic}" means the center of the ith time segment.

Generally, to satisfy the accuracy requirement, the lengths of the time segments should be short, consequently, a large n is necessary. Problem (2) may then have a large number of design variables and constraints.

III. Decomposition Using Collaborative Optimization

CO has a bilevel computational structure in which the original optimization problem is decomposed into one system-level optimization and several suboptimizations. The system-level optimization deals with global design variables shared by more than one suboptimization to optimize the system-level objective and coordinate the inconsistencies among different suboptimizations, whereas each suboptimization deals with local variables to match the target set by the system level and satisfy the local constraints. Properties of CO, including advantages and disadvantages, have been discussed from different aspects in previous literature [8–10]. Updates and extensions of CO have also been developed [11–13].

From the viewpoint of practical computation, a benefit of CO for a large-scale optimization is that the original large number of design variables and constraints are divided into several suboptimizations, each of which is reduced and, consequently, easily dealt with. In the case that the conventional integrated method fails to solve the problem with a large number of design variables and constraints, the decomposed solution may still be an alternative way, which is illustrated by the numerical case in the following section. Moreover, there is the potential of parallel computations for these suboptimizations under the architecture of CO, which means that although the total amount of computation in the decomposed method is generally increased, compared with that in the integrated method, the parallel computations provide the possibility of "trading space for time." Meanwhile, it should be noted that to avoid a high-dimensional system-level problem, CO tends to be most effective in weak-coupled problems, i.e., problems with few global variables [14], and the formulation of problem (2) exactly satisfies this requirement, which will be explained in the following paragraph and illustrated in the numerical example.

Suppose problem (2) is decomposed into two subproblems at the time node t_d, that is, time interval $[t_0, t_d]$ is for subproblem 1 with n_1 segments, whereas $[t_d, t_f]$ is for suboptimization 2 with n_2 segments, $n_1 + n_2 = n$. Consequently, the design variables and constraints at the time nodes in problem (2) are naturally divided into the two subproblems. In $[X^i, Y^i]$ $i = 1, \ldots, n$, the global design variables shared by both subproblems are only those at the dividing time point t_d, i.e., $[X^{n1}, Y^{n1}]$. Additionally, suppose all the static variables $P = [p_1 \ldots p_q]$ are global variables, whereas q, the dimension of P, is generally not a big number. As a result, the converted nonlinear optimization [problem (2)] is a weak-coupled problem with few global variables.

The reformulation of problem (2) under the CO structure is as follows:

Subproblem 1:

$$\begin{gathered}\text{Find } [Y^0 \ldots Y^{n1}, X^0 \ldots X^{n1}, P] \\ \text{Min } \|Y^{n1\,\text{sys}} - Y^{n1}\| + \|X^{n1\,\text{sys}} - X^{n1}\| + \|P^{\text{sys}} - P\|\end{gathered} \tag{3a}$$

$$\text{s.t. } C_s^i\colon\ Y^i - Y^{i+1} + \frac{h^i}{6}(F^i + 4F^{ic} + F^{i+1}) = 0 \quad i = 0, 1 \ldots n_1 - 1 \tag{3b}$$

$$C_t^i\colon\ C_t^i(Y^i, X^i, P) \le 0 \qquad i = 0, 1 \ldots n_1 \tag{3c}$$

$$C_b\colon\ C_b(Y^0, X^0, P) = 0 \tag{3d}$$

Similarly, the formulation of subproblem 2 is

$$\begin{gathered}\text{Find } [Y^{n1} \ldots Y^n, X^{n1} \ldots X^n, P] \\ \text{Min } \|Y^{n1\,\text{sys}} - Y^{n1}\| + \|X^{n1\,\text{sys}} - X^{n1}\| + \|P^{\text{sys}} - P\|\end{gathered} \tag{4a}$$

$$\text{s.t. } C_s^i\colon\ Y^i - Y^{i+1} + \frac{h^i}{6}(F^i + 4F^{ic} + F^{i+1}) = 0 \quad i = n_1, \ldots, n-1 \tag{4b}$$

Table 1 Design variables and constraints in the optimizations

		Design variables	Constraints
Converted nonlinear optimization		$[x^0, \ldots, x^{40}, y_1^0, \ldots, y_1^{40}, y_2^0, \ldots, y_2^{40}, y_3^0, \ldots, y_3^{40}, t_f]$ 165 variables	$[c_{s1}^0, \ldots c_{s1}^{39}, c_{s2}^0, \ldots c_{s2}^{39}, c_{s3}^0, \ldots c_{s3}^{39}, c_{t1}^0, \ldots c_{t1}^{40}, c_{t2}^0, \ldots c_{t2}^{40}, c_{b1}, c_{b2}, c_{b3}, c_{b4}, c_{b5}, c_{b6}]$ 208 constraints
CO	System level	$[x^{20\text{sys}}, y_1^{20\text{sys}}, y_2^{20\text{sys}}, y_3^{20\text{sys}}, t_f^{\text{sys}}]$ Five variables	Two system-level constraints
	Subproblem 1	$[x^0, \ldots, x^{20}, y_1^0, \ldots, y_1^{20}, y_2^0, \ldots, y_2^{20}, y_3^0, \ldots, y_3^{20}, t_f]$ 85 variables	$[c_{s1}^0, \ldots c_{s1}^{19}, c_{s2}^0, \ldots c_{s2}^{19}, c_{s3}^0, \ldots c_{s3}^{19}, c_{t1}^0, \ldots c_{t1}^{20}, c_{t2}^0, \ldots c_{t2}^{20}, c_{b1}, c_{b2}, c_{b3}]$ 105 constraints
	Subproblem 2	$[x^{20}, \ldots, x^{40}, y_1^{20}, \ldots, y_1^{40}, y_2^{20}, \ldots, y_2^{40}, y_3^{20}, \ldots, y_3^{40}, t_f]$ 85 variables	$[c_{s1}^{20}, \ldots c_{s1}^{39}, c_{s2}^{20}, \ldots c_{s2}^{39}, c_{s3}^{20}, \ldots c_{s3}^{39}, c_{t1}^{20}, \ldots c_{t1}^{40}, c_{t2}^{20}, \ldots c_{t2}^{40} c_{b4}, c_{b5}, c_{b6},]$ 105 constraints

$$C_t^i\colon\ C_t^i(Y^i, X^i, P) \le 0 \qquad i = n_1, \ldots, n \tag{4c}$$

$$C_b\colon\ C_b(Y^n, X^n, P) = 0 \tag{4d}$$

The system-level optimization is

$$\text{Find } [Y^{n1\,\text{sys}}, X^{n1\,\text{sys}}, P^{\text{sys}}]$$

$$\text{Min } J = \phi(Y^{n1\,\text{sys}}, X^{n1\,\text{sys}}, P^{\text{sys}}) \tag{5a}$$

$$\text{s.t. } \|Y^{n1\,\text{sys}} - \hat{Y}^{n11}\| + \|X^{n1\,\text{sys}} - \hat{X}^{n11}\| + \|P^{\text{sys}} - \hat{P}^1\| = 0 \tag{5b}$$

$$\|Y^{n1\,\text{sys}} - \hat{Y}^{n12}\| + \|X^{n1\,\text{sys}} - \hat{X}^{n12}\| + \|P^{\text{sys}} - \hat{P}^2\| = 0 \tag{5c}$$

In problem (5), superscripts “^{1}”, “^{2}” and “$^{\text{sys}}$” mean the global design variables in subproblem 1 (problem 3), subproblem 2 (problem 4), and system level (problem 5), respectively. Superscript “ˆ” means the optimum of the variable. In practical computation, the system-level consistency equality constraints (5b–5c) are relaxed into inequality constraints to avoid computational difficulties. Special attention should be paid to this relaxation because there are two different relaxation ways: one way is with relaxed tolerance [9] and the other way is without relaxed tolerance [8]. What we used in this Note is the former way, and the relaxed tolerance is 10^{-5}.

Thus, we can see that the nonlinear optimization problem (2) has been decomposed into three optimization problems, (3–5), each of which has fewer design variables and constraints compared with the original problem.

IV. Numerical Test Case and Discussions

The statement of this case is as follows [5]:

$$\text{Min } t_f$$

$$\text{s.t. } c_{s1}\colon\ \frac{\mathrm{d}y_1(t)}{\mathrm{d}t} - \cos(y_3(t)) = 0$$

$$c_{s2}\colon\ \frac{\mathrm{d}y_2(t)}{\mathrm{d}t} - \sin(y_3(t)) = 0$$

$$c_{s3}\colon\ \frac{\mathrm{d}y_3(t)}{\mathrm{d}t} - x(t) = 0 \qquad c_{t1}\colon\ x(t) \ge 0$$

$$c_{t2}\colon\ x(t) - 1 \le 0 \qquad c_{b1}\colon\ y_1(t_0) = 0 \qquad c_{b2}\colon\ y_2(t_0) = 0$$

$$c_{b3}\colon\ y_3(t_0) + \pi = 0 \quad c_{b4}\colon\ y_1(t_f) = 0 \quad c_{b5}\colon\ y_2(t_f) = 0$$

$$c_{b6}\colon\ y_3(t_f) - \pi = 0 \tag{6}$$

Table 2 Results of integrated solution

	Initial values					
	Set 1	Set 2	Set 3	Set 4	Set 5	Set 6
Optimal objective	6.2832	6.2832	6.2832	Fail	6.2832	Fail
Number of function calls	1660	1494	1992	Fail	4484	Fail

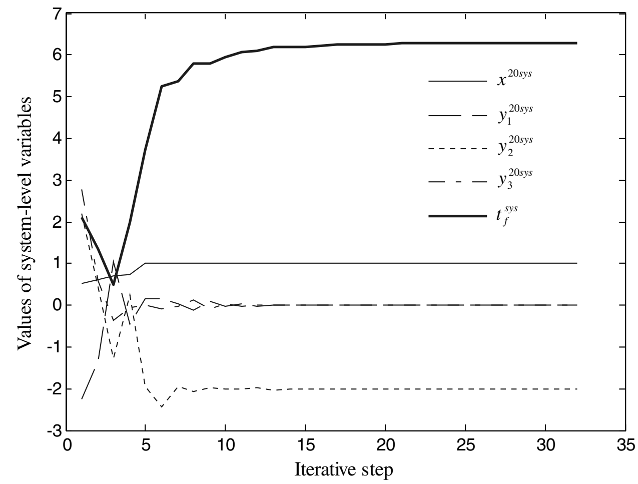


Fig. 1 Iterative history of system-level variables.

The theoretical solution to problem (6) is as follows:

$$\hat{t}_f = 2\pi$$

$$\hat{x}(t) = 1 \qquad \hat{y}_1(t) = -\sin(t)$$

$$\hat{y}_2(t) = -1 + \cos(t) \qquad \hat{y}_3(t) = t - \pi \tag{7}$$

Suppose problem (6) is converted into a nonlinear optimization using collocation with 40 uniformly divided time subintervals. Then the converted problem is decomposed into two subproblems and one system-level problem at the half-time point $t_f/2$. The design variables and constraints in the converted nonlinear optimization and the corresponding optimizations under the CO architecture are listed in Table 1. The upper bounds and lower bounds of the design variables in the converted nonlinear optimization problems are set as follows:

$$0 \le x^0, \ldots, x^{40} < 1$$

$$-5 \le y_1^0, \ldots, y_1^{40}, y_2^0, \ldots, y_2^{40}, y_3^0, \ldots, y_3^{40} < 5$$

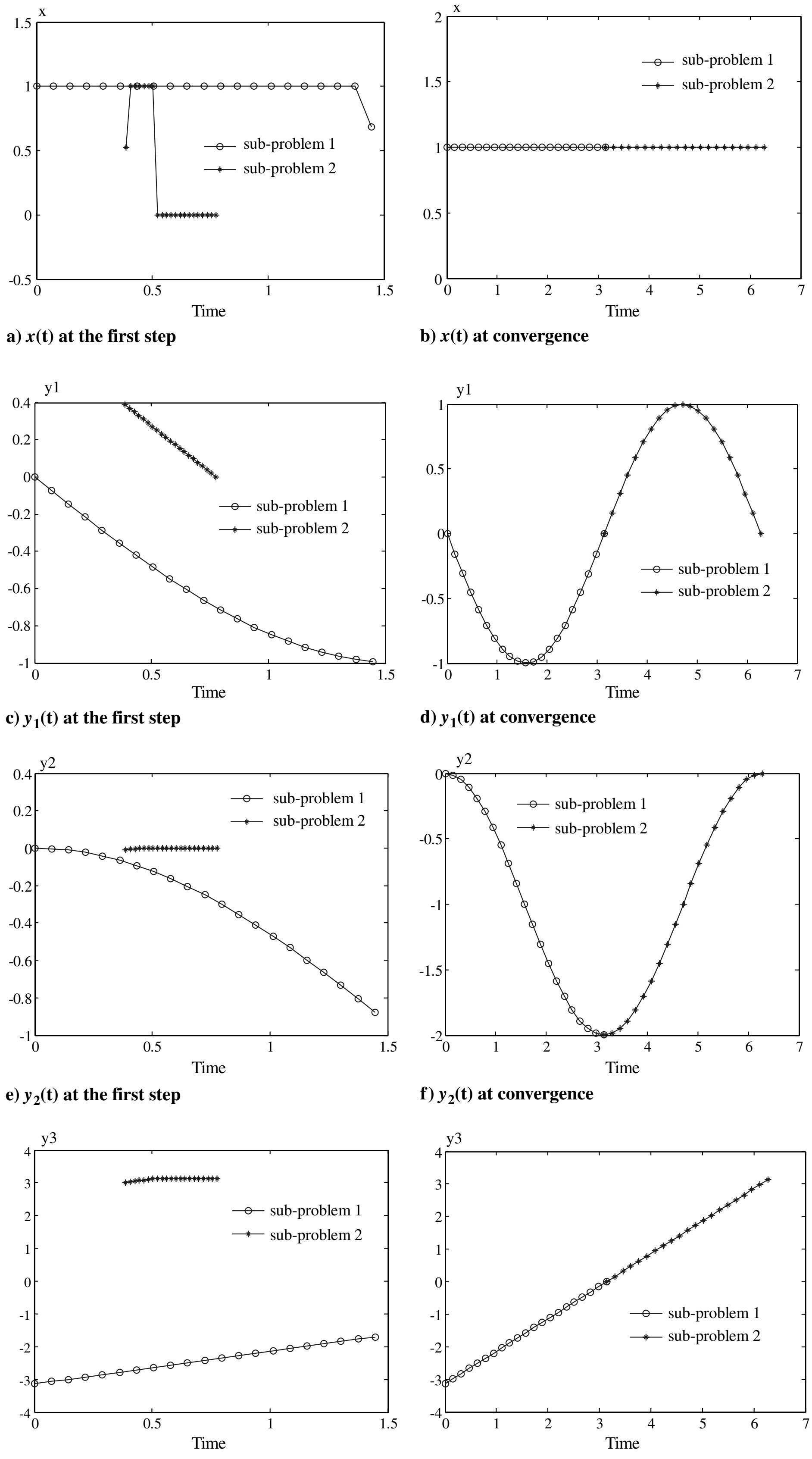
$$0 \le t_f < 10 \tag{8}$$

$[x^{20}, y_1^{20}, y_2^{20}, y_3^{20}]$ at the dividing time point $t_f/2$, as well as the final time t_f, are shared by both subproblem 1 and subproblem 2. So they are global variables and then treated as system-level variables to be optimized. For the simplification of presentation, the formulations of the converted nonlinear optimization problem of problem (6), and the corresponding optimizations in CO are not provided, which can be constructed according to optimization problems (2–5), conveniently.

In this study, the numerical case is solved by using both the conventional integrated solution and the discussed CO solution for comparison. The MATLAB (R2011b) optimization toolbox function “fmincon” with the default settings is employed as the optimizer. In the fmincon function of this version, sequential quadratic

Table 3 Results of CO solution

			Subproblem initial values					
			Set 1	Set 2	Set 3	Set 4	Set 5	Set 6
System-level initial values	Set 1	No. 1	344,864	341,080	307,762	619,205	355,639	353,326
		No. 2	310,982	280,023	295,926	310,060	298,656	315,849
		Avr. no.	327,923	310,552	301,844	464,633	327,148	334,588
		Opt. obj.	6.2778	6.2778	6.2778	6.2778	6.2778	6.2778
	Set 2	No. 1	299,544	285,176	253,080	541,356	310,994	298,455
		No. 2	287,501	246,476	261,870	277,225	260,609	274,113
		Avr. no.	293,522	265,826	257,475	409,291	285,802	286,284
		Opt. obj.	6.2778	6.2778	6.2778	6.2778	6.2778	6.2778
	Set 3	No. 1	313,142	299,022	269,881	579,002	309,127	316,560
		No. 2	301,001	270,470	286,129	302,054	280,018	295,513
		Avr. no.	307,072	284,746	278,005	440,528	294,573	306,037
		Opt. obj.	6.2778	6.2778	6.2778	6.2778	6.2778	6.2778



a) $x(t)$ at the first step
b) $x(t)$ at convergence
c) $y_1(t)$ at the first step
d) $y_1(t)$ at convergence
e) $y_2(t)$ at the first step
f) $y_2(t)$ at convergence
g) $y_3(t)$ at the first step
h) $y_3(t)$ at convergence

Fig. 2 Solutions at the first step and convergence of the two suboptimizations: the time scale of a), c), e), and g) is different from that of b), d), f), and h).

programming (SQP) is one of the four optimization algorithm options, which is used in this Note. We use two ways to select the initial values of the design variables. One way is that all the 165 design variables are randomly selected between the upper bounds and lower bounds; the other way is that the variables with the same title are given the same value, which is also randomly selected between the upper bounds and lower bounds. For example, $x^0, \ldots, x^{40}$, are given the same value between the upper bound and lower bound of x. The

MATLAB function "rand", as well as the upper and lower bounds of the design variables are used to generate these initial values. Six sets of initial values are generated: sets 1–3 are in the former way, and sets 4–6 are in the latter way.

The results of the integrated solution and the CO solution are listed in Tables 2 and 3, respectively. Note that, as shown in Table 1, there are three optimizations in CO, i.e., one system-level optimization and two suboptimizations. As a result, we should select three sets of initial values for these three optimizations, respectively. Each of the six sets of initial values used in the integrated solution is decomposed into two parts according to Table 1 to serve as the initial values of the two suboptimizations, respectively, which can make the comparison more reasonable because it makes the integrated solution and the CO solution from the same initial values. Additionally, three sets of initial values of system-level variables $[x^{20\mathrm{sys}}, y_1^{20\mathrm{sys}}, y_2^{20\mathrm{sys}}, y_3^{20\mathrm{sys}}, t_f^{\mathrm{sys}}]$ are randomly generated between the upper bounds and lower bounds.

For the simplification of presentation, the initial values of sets 1–3 are not listed, and sets 4–6 are listed as follows:

Set 4: $[x^0, \ldots, x^{40}] = 0.7403$ $[y_1^0, \ldots, y_1^{40}] = -2.4583$ $[y_2^0, \ldots, y_2^{40}] = 3.4558$ $[y_3^0, \ldots, y_3^{40}] = 0.3815$ $t_f = 9.1014$

Set 5: $[x^0, \ldots, x^{40}] = 0.3507$ $[y_1^0, \ldots, y_1^{40}] = 4.3532$ $[y_2^0, \ldots, y_2^{40}] = 4.2772$ $[y_3^0, \ldots, y_3^{40}] = 0.8770$ $t_f = 1.0176$

Set 6: $[x^0, \ldots, x^{40}] = 0.3669$ $[y_1^0, \ldots, y_1^{40}] = -2.2415$ $[y_2^0, \ldots, y_2^{40}] = -2.3390$ $[y_3^0, \ldots, y_3^{40}] = -2.7346$ $t_f = 0.0132$

The three sets of initial values of system-level variables $[x^{20\mathrm{sys}}, y_1^{20\mathrm{sys}}, y_2^{20\mathrm{sys}}, y_3^{20\mathrm{sys}}, t_f^{\mathrm{sys}}]$ are as follows:

Set 1: $[\,0.8930 \quad -0.4650 \quad 0.7842 \quad -1.8439 \quad 9.9402\,]$

Set 2: $[\,0.9839 \quad 4.6455 \quad 1.6627 \quad 2.2616 \quad 3.3404\,]$

Set 3: $[\,0.5228 \quad -2.2665 \quad 2.1838 \quad 2.7802 \quad 2.1064\,]$

In Table 3, "No. 1" and "No. 2" mean the numbers of function calls in suboptimization 1 and suboptimization 2, respectively. Below that, "Avr. no." means the average of the numbers of function calls in these two suboptimizations. Note that the number of constraint functions in each of the two suboptimizations is only half of that in the original integrated formulation, so Avr. no., rather than the sum of the numbers of the function calls in the two suboptimizations should be used for the evaluation of the amount of computation. The objective value of CO is slightly smaller than that of the integrated solution, which is due to the allowed relaxed tolerance (10^{-5}) of the system-level constraints.

For the details of the CO solution, we take the computation with set 3 system-level initial values and set 6 suboptimization initial values, for example, the results of which are illustrated in Figs. 1 and 2.

At the beginning of CO, the initial values of the system-level variables (i.e., the system-level target) are far from the theoretical values, consequently, there are inconsistencies among the target and the optimal solutions of the two suboptimizations. For example, $t_{\mathrm{sys}}|_1 = 2.1064$, $\hat{t}_f^1|_1 = 1.4465$, $\hat{t}_f^2|_1 = 0.7757$, which results in a time overlapping between the solutions of the two suboptimizations, shown in Figs. 2a, 2c, 2e, and 2g. At convergence (the 32nd step), the suboptimization 1 solution and the suboptimization 2 solution are consistent (within the relaxed tolerance) and both agree well with the theoretical results, as shown in Figs. 2b, 2d, 2f, and 2h. For example, $\hat{t}_f^1|_{32} = 6.2795$, $\hat{t}_f^2|_{32} = 6.2795$, whereas the system-level-optimal value is $\hat{t}_f^{\mathrm{sys}}|_{32} = 6.2778$. Note that there are slight inconsistencies between the system-level target and the responses of the suboptimizations, which are also due to the relaxed tolerance in the system-level constraints.

The comparison between the integrated solution and the CO solution to this case can be made from two aspects, i.e., the amount of computation and the robustness of computation. For the amount of computation, the results in this Note are consistent with those in the previous reports that CO needs much more function calls to achieve convergence [15,16]. Figure 1 can, to some extent, explain this phenomenon, that is, the system-level convergence rate near the optimum is very slow, which is an intrinsic drawback of CO. However, for the robustness of computation, the results in this Note are somewhat different from those in the previous reports. The results show that in the integrated solution, two of the six initial values fail to find a result, whereas all of the CO attempts successfully converge to the correct result. In contrast, in the previous reports [9,10,15,16], CO is generally thought to be a less robust MDO method. We think that a possible explanation of this phenomenon can be summarized in two aspects. One aspect is that, as discussed in Sec. III, the formulation of the large-scale optimization problem converted from an optimal control problem exactly satisfies the requirements of CO. The other aspect is that the comparison in this Note and that in the previous reports are made in different ways. In [15,16], the MDO methods are applied to several different problems, and the computational conditions are not provided whereas in this Note, the integrated solution and the CO solution are applied to only one problem, and the comparison is made for different computational conditions. We think that the comparison way in this Note is more detailed than that in the previous reports.

In addition, there are only two equal-scale subproblems in this Note for the simplification of presentation, whereas more subproblems are permitted in the architecture of CO, which means that designers can determine the scale of each subproblem according to the actual situation of the original large-scale optimization problem. Moreover, research on the optimal partitioning can be used to guide the decomposition [17].

V. Conclusions

The nonlinear optimization problem with a large number of design variables and constraints converted from an optimal control problem is a weak-coupled problem, which is suitable to apply collaborative optimization (CO) for a solution. From the viewpoint of practical computation, the decomposition strategy in CO is a natural and potential way for solving this kind of large-scale optimization problem, in which the original large-scale problem is decomposed into several relatively small subproblems.

Work in this Note is a preliminary attempt to apply multidisciplinary design optimization (MDO) methods to optimal control problems, the purpose of which is to find an alternative solution for this kind of problem. Many efforts are still necessary, for example, improving CO for higher efficiency, applying CO to more complex engineering optimal control problems, and investigating the effects of the applications of other decomposition-based MDO methods to optimal control problems.

Acknowledgement

This research is supported by the National Natural Science Foundation of China (50875024,11202024) and the Aeronautical Science Foundation of China (2009ZA72002).

References

[1] Paiva, R. M., Carvalho, A. R. D., Crawford, C., and Suleman, A., "Comparison of Surrogate Models in a Multidisciplinary Optimization Framework for Wing Design," *AIAA Journal*, Vol. 48, No. 5, 2010, pp. 995–1006. doi:10.2514/1.45790

[2] Zhang, K. S., Han, Z. H., Li, W. J., and Song, W. P., "Coupled Aerodynamic/Structural Optimization of a Subsonic Transport Wing Using a Surrogate Model," *Journal of Aircraft*, Vol. 45, No. 6, 2008, pp. 2167–2170. doi:10.2514/1.36047

[3] Betts, J. T., *Practical Methods for Optimal Control and Estimation Using Nonlinear Programming*, Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA, 2009, pp. 91–122.

[4] Hull, D. G., "Conversion of Optimal Control Problems into Parameter Optimization Problems," *Journal of Guidance, Control, and Dynamics*, Vol. 20, No. 1, 1997, pp. 57–60. doi:10.2514/2.4033

[5] Darby, C. L., Hager, W. W., and Rao, A. V., "Direct Trajectory Optimization Using a Variable Low-Order Adaptive Pseudospectral Method," *Journal of Spacecraft and Rockets*, Vol. 48, No. 3, 2011, pp. 433–445. doi:10.2514/1.52136

[6] Gill, P. E., Murray, W., and Saunders, M. A., "SNOPT: An SQP Algorithm for Large-Scale Constrained Optimization," *SIAM Review*, Vol. 47, No. 1, 2005, pp. 99–131. doi:10.1137/S0036144504446096

[7] Wang, D. P., Gary, W. G., and Naterer, G. F., "Collaboration Pursuing Method for Multidisciplinary Design Optimization Problems," *AIAA Journal*, Vol. 45, No. 5, 2007, pp. 1091–1103. doi:10.2514/1.19472

[8] Braun, R. D., Gage, P., Kroo, I. M., and Sobieski, I., "Implementation and Performance Issues in Collaborative Optimization," *6th AIAA/NASA/ISSMO Symposium on Multidisciplinary Analysis and Optimization*, AIAA Paper 1996-4017, 1996.

[9] Alexandrov, N. M., and Lewis, R. M., "Analytical and Computational Aspects of Collaborative Optimization for Multidisciplinary Design," *AIAA Journal*, Vol. 40, No. 2, 2002, pp. 301–309. doi:10.2514/2.1646

[10] DeMiguel, A. V., and Murray, W., "An Analysis of Collaborative Optimization Methods," *8th AIAA/USAF/NASA/ISSMO Symposium on Multidisciplinary Analysis and Optimization*, AIAA Paper 2000-4720, 2000.

[11] Sobieski, I. P., and Kroo, I. M., "Collaborative Optimization Using Response Surface Estimation," *AIAA Journal*, Vol. 38, No. 10, 2000, pp. 1931–1938. doi:10.2514/2.847

[12] Roth, B., and Kroo, I., "Enhanced Collaborative Optimization: Application to an Analytic Test Problem and Aircraft Design," *Proceedings of the 12th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference*, AIAA Paper 2008-5841, 2008.

[13] Su, R. Y., Gu, L. J., and Fan, Z. J., "Multi-Objective Collaborative Optimization Based on Evolutionary Algorithms," *Journal of Mechanical Design*, Vol. 133, No. 10, 2011, p. 104502. doi:10.1115/1.4004970

[14] Tedford, N. P., and Martins, J. R. R. A., "Benchmarking Multidisciplinary Design Optimization Algorithms," *Optimization and Engineering*, Vol. 11, No. 1, 2010, pp. 159–183. doi:10.1007/s11081-009-9082-6

[15] Yi, S. I., Shin, J. K., and Park, G. J., "Comparison of MDO Methods With Mathematical Examples," *Structural and Multidisciplinary Optimization*, Vol. 35, No. 5, 2008, pp. 391–402. doi:10.1007/s00158-007-0150-2

[16] Price, A. R., and Keane, A. J., "On the Coordination of Multidisciplinary Design Optimization Using Expert Systems," *AIAA Journal*, Vol. 49, No. 8, 2011, pp. 1778–1794. doi:10.2514/1.J050928

[17] Allison, J. T., Kokkolaras, M., and Papalambros, P. Y., "Optimal Partitioning and Coordination Decisions in Decomposition-Based Design Optimization," *Journal of Mechanical Design*, Vol. 131, No. 8, 2009, p. 081008. doi:10.1115/1.3178729

J. Martins
Associate Editor