EVALUATION AND IMPLEMENTATION OF MULTIDISCIPLINARY DESIGN OPTIMIZATION STRATEGIES

Guillaume Renaud and Guoqin Shi
Institute for Aerospace Research
National Research Council Canada
Ottawa, Canada K1A 0R6

Keywords: *MDO, MDF, IDF, CO, CSSO, BLISS*

Abstract

Several MDO formulations have been proposed in the literature to overcome the difficulties in optimizing coupled engineering systems. Some of these formulations were evaluated and compared in this paper to determine which should be envisaged for establishing a practical MDO capability. The methods were tested using two demonstrative examples taken from the literature. The results showed important differences in the behavior of each method, especially concerning reliability and efficiency.

1. Introduction

Multidisciplinary design optimization (MDO) is a methodology developed to address the computational and organizational complexity of complicated engineering systems. The objective of a MDO process is to determine a design that optimizes certain system performance measures while considering the effects of several mutually interactive disciplines. Conceptually, many approaches can be taken to address this type of problem.

Several MDO formulations have been proposed in recent years. Each has its own way of decomposing the problem into subtasks, resulting in specific characteristics related to ease of implementation and use, complexity, robustness, and computational efficiency. Moreover, some of these formulations permit the use of autonomous concurrent disciplinary optimizations allowing parallel computational work for each discipline. With these autonomous decomposition approaches each team of disciplinary experts can be in charge of its own aspects of the project and be totally free to choose the set of methods that are best suited for its particular subsystem [1].

In this study, five existing MDO methods are described, evaluated and compared: the MultiDisciplinary Feasible (MDF) method [2], the Individual Discipline Feasible (IDF) method [3], the Concurrent SubSpace Optimization with Response Surfaces (CSSO/RS) method [4], the Collaborative Optimization (CO) method [5], and the Bi-Level Integrated System Synthesis with Response Surfaces (BLISS/RS) method [6].

The evaluation presented in this paper has been entirely performed using MATLAB and its optimization toolbox [7,8].

2. Multidisciplinary design analysis and optimization

In general, three types of variables can be defined in a multidisciplinary problem. The first type corresponds to the *global* or *shared* input variables z, required by more than one discipline, or by system-level calculations. Conversely, *disciplinary* or *local* input variables x_i are used in calculations concerning the i^{th} discipline only. Finally, *state* or *behavior* variables y_i, corresponding to the disciplinary responses, are subsystem output values that can be used as input parameters to other disciplinary calculations as, for a N-discipline problem;

$$y_i = y_i(x_i, y_j, z), \quad j = 1,\ldots,N, \quad j \neq i.$$

The analysis of a multidisciplinary system consists of determining the disciplinary state parameters y that correspond to a specific set of local and shared input parameters x and z. The conventional solution approach results in a sequence of iterations between the various disciplinary analyses. At convergence, all state variables are compatible and the system is in a so-called state of *multidisciplinary feasibility*. A typical multidisciplinary analysis is sketched in Figure 1 for a three-discipline system.

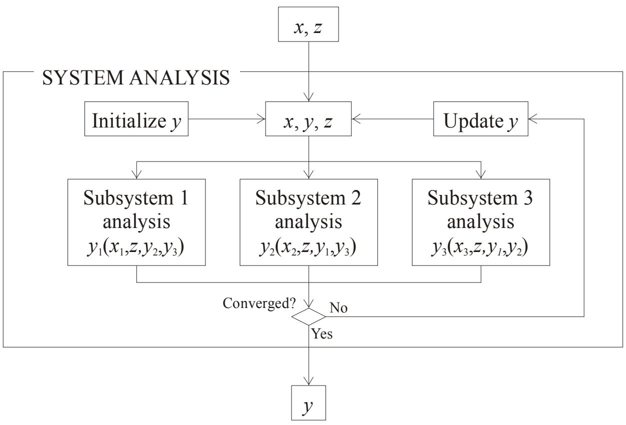

Figure 1. Multidisciplinary analysis.

The optimization of such a system is the process of determining the values of x and z, within certain limits, that minimize or maximize an objective function $f(z, y(x, y, z))$ representing the system performance potentially subject to some global constraints. Further, each subsystem is subject to local constraints, g_i. To be acceptable, the final design must be multidisciplinary feasible, must satisfy all constraints, and must utilize the same values for the shared variables z in all subsystems.

As was mentioned, there exist several approaches to formulate MDO. Each approach is based on different concepts, which result in a certain level of complexity and efficiency. The formulations consist of sequences of subtasks including disciplinary analyses, disciplinary and system sensitivity analyses, optimization and approximations at the subsystem (discipline) and system (coordination) levels.

3. MDO Formulation

This section briefly describes the five MDO formulations that are evaluated in this study.

The **MultiDisciplinary Feasible (MDF)** or **All-in-One (A-i-O)** [2] method is the simplest way to perform MDO. The multidisciplinary problem is fully solved by conventional iterative methods at each optimization step resulting in a system that is always multidisciplinary feasible. It is not a decomposition method and does not exploit the modularity of the problem. Therefore, all variables and constraints are treated at the global level.

The MDF optimization can be stated as;

$$\begin{array}{ll} \text{minimize} & f(z, y(x, y, z)) \\ \text{subject to} & g(z, y(x, y, z)) \leq 0 \end{array}$$

where f is the objective function and g represents all system and/or disciplinary constraints. Figure 2 demonstrates this method.

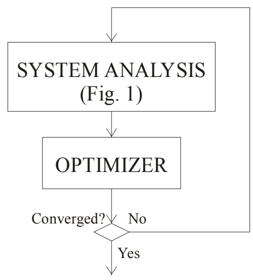

Figure 2. MDF method.

The **Individual Discipline Feasible (IDF)** [2][3] method uncouples the disciplinary analyses but keeps the optimization at the system level to form a single-level decomposed optimization problem. The subsystems are individually analyzed and the optimization is performed for the system as a whole with constraints imposing multidisciplinary feasibility using extra coupling variables that are introduced in the formulation. The disciplines are always feasible, individually, but the complete system may not be feasible until the optimization process converges.

The IDF optimization statement is;

$$\begin{aligned} &\text{minimize} && f(z, y(x, y', z)) \\ &\text{subject to} && g(z, y(x, y', z)) \leq 0 \\ & && y' - y(x, y', z) = 0 \end{aligned}$$

where y' are auxiliary disciplinary input variables corresponding to the various state variables. The second constraint ensures multidisciplinary feasibility when y is equal to y'. The IDF procedure is illustrated in Figure 3.

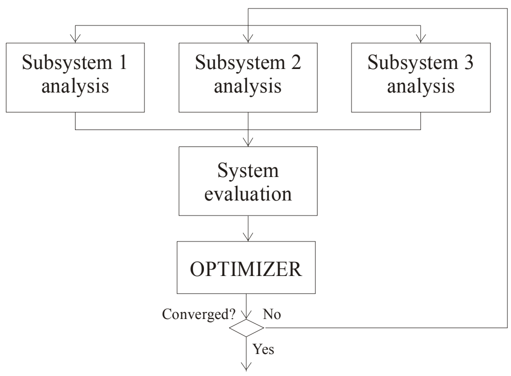

Figure 3. IDF method

The **Collaborative Optimization (CO)** [2],[5],[9],[10],[11], which decomposes and reformulates the problem as a bi-level optimization, is one of the most studied and documented MDO formulations. The system is optimized at the coordination level by determining target values for subsystem responses and shared design variables with compatibility constraints that ensure multidisciplinary feasibility. The optimization objectives for the subsystems are to match, as closely as possible, these target values while satisfying local disciplinary constraints. As in the IDF method, multidisciplinary feasibility is achieved at the end of the process. If the target values corresponding to the shared and state variables are z and y', respectively, the system level optimization problem can be written as;

$$\begin{aligned} &\text{minimize} && f(z, y') \\ &\text{subject to} && c(z, z^*, y', y(x_i^*, y', z_i^*)) = 0 \end{aligned}$$

where c represents the compatibility constraints, one for each subsystem, of the form;

$$c_i = (z - z_i^*)^2 + (y' - y_i(x_i^*, y', z_i^*))^2$$

where the asterisks indicate optimal subsystem values.

Similarly, at the disciplinary level, the i^th^ subsystem optimization can be stated as;

$$\begin{aligned} &\text{minimize} && c_i(z, z_i, y', y(x_i, y', z_i)) \\ &\text{subject to} && g(x, z, y(x, y, z)) \leq 0 \end{aligned}$$

where the objective c_i is of the same form as the constraints at the global level.

The CO procedure is illustrated in Figure 4.

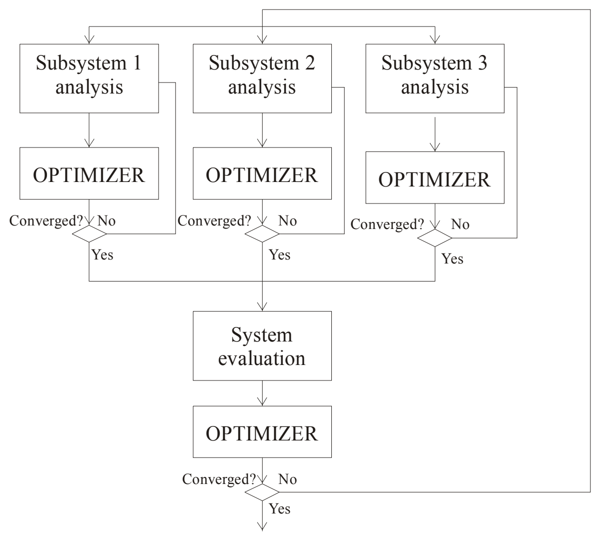

Figure 4. CO method.

The **Concurrent SubSpace Optimization with Response Surfaces (CSSO/RS)** [4] is also a true decomposition strategy. The system and subsystems are optimized sequentially using their specific objective functions, constraints, and variables. A specific system performance is approximated in each subsystem optimization using Response Surface (RS) models for simulating other discipline state variables. Similarly, the system is optimized at the coordination level using RS models, which replace the required disciplinary analyses. As the analyses and optimizations are performed, more information is known about the actual system, thus permitting updates to the various models. The i^{th} discipline optimization can be stated as;

$$\begin{array}{ll} \text{minimize} & f(z, y_i(x_i, y_j^{app}, z_i), y_j^{app}) \\ \text{subject to} & g_i(x_i, z, y_i(x_i, y_j^{app}, z_i), y_j^{app}) \leq 0 \end{array}$$

where $y_j^{app} = y_j^{app}(z, x_j)$ represents the other discipline approximate state responses. A complete multidisciplinary analysis is performed for each subsystem optimal design to generate a set of multidisciplinary feasible designs that will improve the quality of the approximation models. A system optimization is then performed as;

$$\begin{array}{ll} \text{minimize} & f(z, y^{app}) \\ \text{subject to} & g(z, y^{app}) \leq 0. \end{array}$$

Another multidisciplinary analysis is performed with the system optimal design to further improve the models, and the whole process is repeated until convergence, as shown in Figure 5.

Finally, in the **Bi-Level Integrated System Synthesis with Response Surfaces (BLISS/RS)** [6] each subsystem is optimized with respect to local variables holding the shared variables constant to minimize the system objective under local constraints. The shared variables are utilized by the system level optimization, only. Total derivatives [12] are used to predict the effects of each set of variables on the objective function. The optimization of the i^{th} discipline takes the form;

$$\begin{array}{ll} \text{minimize} & D(f, x_i)^T \Delta x_i \\ \text{subject to} & g_i(x_i) \leq 0 \end{array}$$

where $D(f, x_i)^T$ is the total derivative of the objective function with respect to the local variables of the disciplines. It includes the indirect effects of these variables on other subsystems. The term $D(f, x_i)^T \Delta x_i$ corresponds to the first order predicted objective function change due to a change in x_i. The optimization at the system level takes the same form with the shared variables, using RS approximation models for system performance estimation. A simplified flowchart of the BLISS/RS is presented in Figure 6.

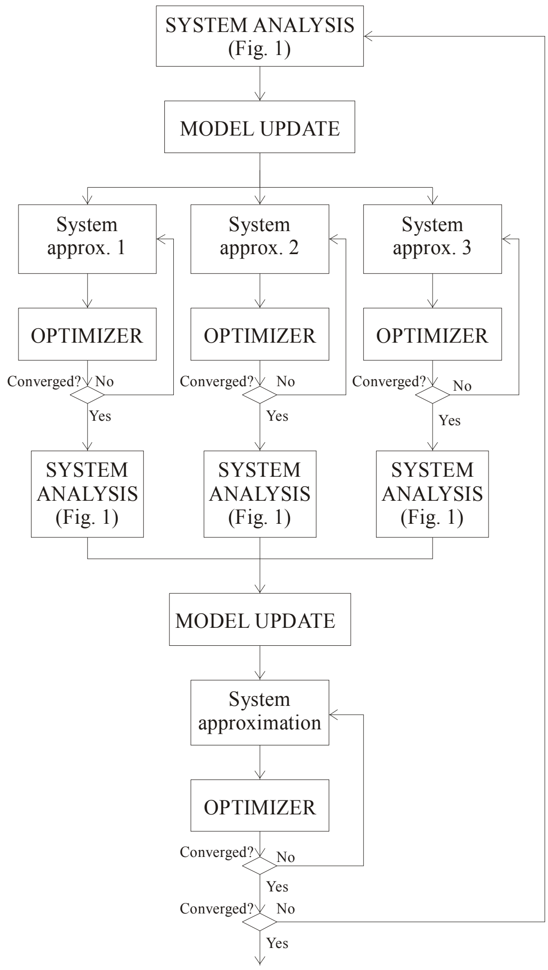

Figure 5. CSSO/RS method.

It should be noted that the original versions of CSSO [13] and BLISS [14-16] do not include Response Surface methodologies. This addition is assumed to be particularly efficient. Indeed, such approximations usually improve the convergence characteristics if the number of parameters used to build the approximation stays limited. BLISS/RS is particularly promising because the number of system variables is kept low resulting in an effective use of such an approximation method.

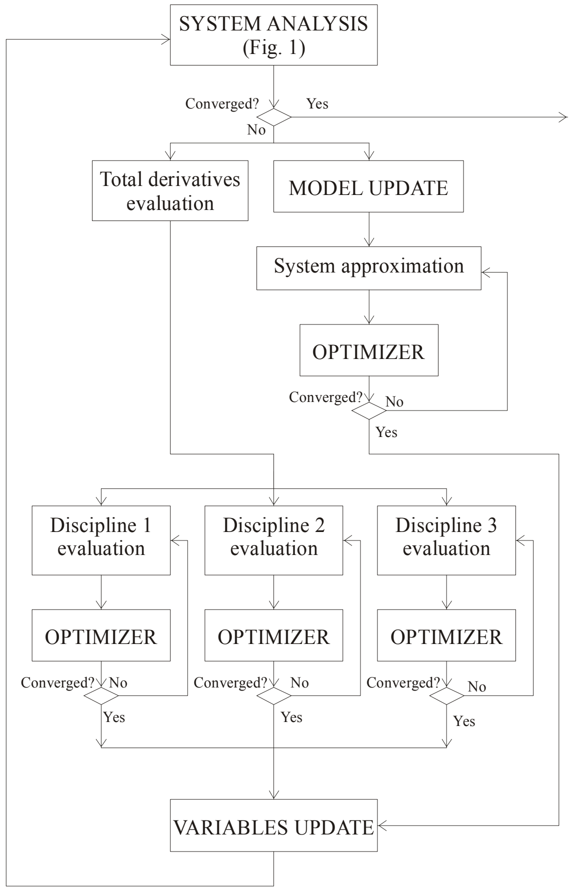

Figure 6. BLISS/RS method.

4. Evaluation

Two examples were implemented to test these five formulations. The efficiency of each method was evaluated by the number of subsystem evaluations required for convergence.

4.1. Example 1: Simple two-discipline problem

The first optimization problem corresponded to the test case that was used to demonstrate the capabilities of CSSO/RS [4]. It consisted of two simple subsystems coupled through shared and state variables. The problem can be stated as;

minimize

$$f = x_2^2 + x_3 + y_1 + e^{-y_2}$$

subject to the constraints

$$g_1 = \left(\frac{y_1}{y_{1a}}\right) - 1 \geq 0$$

$$g_2 = 1 - \left(\frac{y_2}{y_{2a}}\right) \geq 0$$

and to the side constraints

$$-10 \leq x_1 \leq 10$$

$$0 \leq x_2 \leq 10$$

$$0 \leq x3 \leq 10$$

and where the state variables y_1 and y_2, evaluated in subsystems 1 and 2, are defined as

$$y_1 = x_1^2 + x_2 + x_3 - 0.2 y_2$$

$$y_2 = y_1^{1/2} + x_1 + x_3.$$

The y_{1a} and y_{2a} parameters were not given in the aforementioned reference. The values y_{1a} = 8 and y_{2a} = 24 were used since they resulted in the same solution in this paper.

Implementation

Several decisions were taken concerning the implementation of the methods for solving this problem. The following points were adopted for all methods:

i. All three input variables were treated as global variables. Indeed, they were either used in both disciplines (x_1, x_3), or used in the global objective function (x_2, x_3). Consequently, there was no local variable vector x_i.
ii. The termination criterion for multi-disciplinary analysis (Fig. 1) is that the change in each y_i between two consecutive iterations must be lower or equal to 0.0001.
iii. The global objective function was evaluated outside the disciplines at the coordination level.

iv. The upper and lower bounds for the state variables (IDF, CO) were set to positive and negative infinity, respectively. This avoided any beforehand estimation of the final values for these variables.
v. No scaling of variables was made.
vi. The initial values of y were obtained, when required, through a full multidisciplinary analysis. The number of disciplinary analyses needed was included in the total number of subsystem evaluations.

Two formulations were used to test the IDF optimization method . The differences between the formulations lie in the way the y' additional state variables were forced to converge to the actual state variables. Table 2 describes the constraint formulations.

Method	Constraint formulation
IDF 1	$y1 - y1' = 0$ $y2 - y2' = 0$
IDF 2	$(y1 - y1')^2 \leq 0$ $(y2 - y2')^2 \leq 0$

Table 1. IDF formulations.

Four formulations were used to test CO. As with IDF, equality and inequality constraints were tested, as well as analytical gradients instead of finite differences for system constraints. Table 3 describes the constraint formulations.

Method	Constraint formulation
CO 1	$c \leq 0$; finite differences
CO 2	$c \leq 0$; analytical
CO 3	$c = 0$; finite differences
CO 4	$c = 0$; analytical

Table 2. CO formulations.

It is possible to use several techniques to build the model approximations required in the CSSO. The chosen method corresponds to the response surface method proposed in reference [6].

BLISS/RS is not included in this particular evaluation, as there were no local variables x.

The number of iterations to meet convergence is shown in . The red bars correspond to the runs that converged to the wrong solutions.

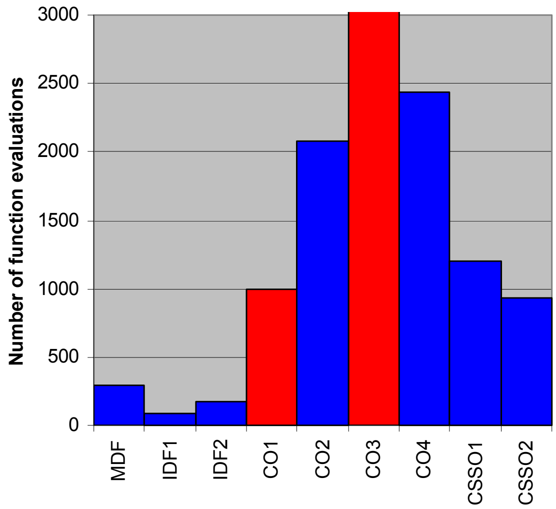

Figure 7: Results for Example 1.

Although most methods resulted in the same solution, the formulations show substantial differences in terms of their numerical efficiencies. For this study, the most efficient method was IDF1 with only 88 disciplinary analyses.

Furthermore, the Collaborative Optimization (CO) based on finite differences failed to converge in one case, and converged to a wrong solution in another. It is also seen that even when this method worked, it was the slowest formulation to converge with at least 2075 disciplinary analyses.

4.2. Example 2: Supersonic business jet preliminary design (four disciplines)

The second optimization problem corresponded to the problem used by NASA to present BLISS [14]. It addressed the maximization of the range of a supersonic business jet. Four coupled subsystems representing structures, aerodynamics, propulsion, and range by semi-empirical relations were used to determine the multidisciplinary state of the aircraft. The first three disciplines are fully coupled since they share common variables and exchange computed values, and the forth discipline receives information from the others to evaluate the performance of the design.

The various disciplines include 6 shared variables, 4 local variables, and 9 disciplinary state variables that act as input variables to other disciplines. The performance value is treated as a 10^{th} state variable evaluated by the range discipline.

It should be noted that there are a number of discrepancies between the equations or references [6, 14-16] and the program presented at the end of reference [14]. Furthermore, an error was reported [17] in the program itself.

A number of parameters were calculated using polynomial functions "*pf*", which are given in the referenced program [14]. The independent variables of these polynomials, given in the parentheses, were normalized by their initial values. However, the initial values that corresponded to the disciplinary state variables are not mentioned in any paper. Based on the fact that the system responses for the initial design must be equal to the initial values of y, a simple iterative sequence of multidisciplinary analyses was performed to determine the initial values of these variables.

Implementation

Due to the nature of the problem, the methods were not implemented exactly as for the first demonstrative example. The following considerations concern all methods:

i. The termination criterion for multidisciplinary analysis (Fig. 1) is that the change in each y_i between two consecutive iterations must be lower or equal to 0.0001.
ii. The global objective function was evaluated inside one of the disciplines.
iii. All variables were scaled using their initial values for all calculations.
iv. When needed, the initial values of y were obtained through a full multidisciplinary analysis before the actual optimization was started. The required disciplinary analyses were included in the total number of subsystem evaluations.

The two variations of IDF were tested. To simulate absolute ignorance of the expected state variables, the upper and lower bounds for the y design variables were set to positive and negative infinity, respectively.

The BLISS approximation models were as described in the CSSO/RS section of problem 1. Also, the g_2 constraint was calculated at the system level since it depends solely on the global variables

The results are shown in Figure 8. In this case, the purple bar corresponds to a method that is unreliable, as described below.

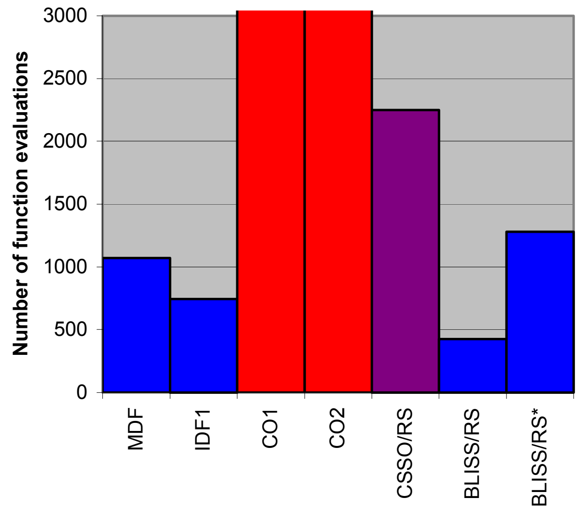

Figure 8: Results for Example 2.

As for Example 1, MDF and IDF1 appear to be computationally efficient. However, as pointed out in the next section, they are probably not the best choice for establishing a practical MDO capability because of their lack of modularity.

The IDF2 method failed because the variable representing the total weight became negative. However, IDF1 performed very well, as shown in Figure 8.

The four variations of CO were tested. Unlike IDF, this method did not behave well when the bounds on the *y* variables were set to large numbers. Even for a restricted design space, this method either failed or was unable to find the true solution. For example, the total weight became less than the fuel weight when using CO1, and CO3 was not able to converge to a solution within the default maximum number of iterations allowed in MATLAB. Many different variations were tested, including changes on the bounds, addition of constraints, and reformulation of existing constraints. None led to better results.

The CSSO/RS method did not appear to be well suited for this particular problem. The response surface approximation used for problem 1 appeared to be effective but not reliable, as most attempts caused premature mathematical problems as encountered with IDF2 and CO1.

Finally, BLISS/RS was able to get the right solution in the smallest number (425) of iterations. Furthermore, it did appear to be reasonably insensitive to the approximation models. Several tests using different models and initial points all converged in less than 1000 disciplinary analyses.

5. Discussion

The two examples showed that the methods differ greatly with respect to their efficiency and reliability.

Several remarks and recommendations concerning the tested methods can be made, both from the numerical tests and from the experience gained in their implementation and use. Obviously, the demonstrative problems included in this study might not reflect the characteristics of large-scale problems. This is why caution should be applied when extrapolating these recommendations to real problems.

Contrary to what is usually stated in the literature, MDF appears to be one of the most efficient approaches, while being the simplest one to implement and use. It is a method worth considering when decomposition and multiprocessing are not desired or possible.

One of the most efficient methods was undoubtedly IDF1, for both tested problems. It was shown to be very stable and reliable since no predetermined bounds were necessary for the state variables. It is also very simple to implement and use, although extra variables are needed to model the state variables. The main problem with this method is that a convergence of the process is needed to ensure multidisciplinary compatibility.

Although these two methods were shown to work well with these two simple problems, care should be taken when applying them to large-scale MDO. For instance, a common optimization method has to be defined for all disciplines. Furthermore, all variables are treated simultaneously, which could lead to less efficient processes as the number of variables increases. Similarly, although worth considering, IDF could be much less efficient when the discipline outputs are in the form of fields represented by a large number of behavior variables

The other three methods offer the type of decomposition that allows the use of specific methods for each discipline.

The CO approach was shown to be ineffective and unreliable. This behavior has been previously reported in the literature. For example, CO has been qualified as “inherently difficult to solve by means of software intended for conventional, single-level, nonlinear programming problems” [2]. It is reported in the same reference that “much fine-tuning would be required to implement the method for a specific problem and that convergence behavior of conventional optimization methods applied to the CO formulation might be erratic”. Characteristics of the CO formulation that explains its “erratic behavior” have also been identified [10].

The CSSO method appeared to be a better option than CO for decomposing the problem. It was however not efficient in solving problem 2 due to mathematical difficulties in the equations. This method should be further tested with other approximation models. Another problem with CSSO is that all the variables are used at the system level. This strategy could lead to very expensive approximation models in a real-problem situation.

Finally, BLISS/RS demonstrated the best ability to solve problem 2 in a fully decomposed manner. It also possesses the desirable characteristics of generating a multidisciplinary compatible design at each iteration cycle, and dealing with x and z design variables only. It is for these reasons that BLISS/RS appears to be the most promising formulation for implementing true MDO capability. Care should be taken in the formulation of the approximation models, as it could significantly affect the overall behavior of the method, especially when solving more realistic problems.

6. BLISS/RS with new approximation model

The good behavior of BLISS/RS with the aforementioned implementation can be partly explained by two facts. First, no discipline calculations had to be performed in the aerodynamics discipline. Indeed, the aerodynamics constraint depends only on z. Second, it was assumed that it is possible to calculate this constraint outside the aerodynamics discipline, at the system level. This assumption may not be valid in a real-problem situation.

A more realistic way of defining the problem is to use a response surface approximation to model the aerodynamics constraint at the system level. The actual constraint values that are used to build this model are calculated at the disciplinary level during the various system analyses.

The BLISS/RS method was unable to converge to a solution with a RS constraint approximation formulated as proposed in reference [6]. This could be due to the fact that the system objective and constraints were approximated in the z space only while they are, in general, functions of both z and x. For example, the design at two distinct iterations can have quite different objective and constraint values even though they have the same or similar z vectors. This can cause excessive distortion in the approximation models, where the approximated functions must vary significantly over a short distance. Furthermore, the objective and constraint values that were used to build the initial linear model may not be representative at subsequent iterations, where x is updated, even if the z values are similar.

An extra step was added to the response surface building procedure to improve the overall behavior of BLISS/RS when faced with approximated global constraints. The idea was to make sure that none of the already stored points were in the neighborhood (in the z space) of the new point that was to be added to the approximation model. If so, the old point was removed and replaced with the new point. This procedure ensured that the approximation model used the most recent information for the current design space region.

The modifications to the BLISS/RS method was more reliable when approximated global constraints were used. Furthermore, the quadratic approximation led to the best results, reported in Figure 8 as “BLISS/RS*”.

7. Conclusion

Five MDO formulations were implemented and tested with two demonstrative problems in a MATLAB environment. The results indicated that substantial efficiency and reliability differences exist among the different methods.

The tests performed showed that three of the methods are worth considering when establishing a true MDO capability. The methods are the MultiDisciplinary Feasible (MDF) method, which corresponds to the traditional "all-in-one" approach, the Individually Feasible (IDF) method, and the Bi-Level Integrated System Synthesis with Response Surfaces (BLISS/RS) method. Additional larger scale work should be carried out to determine if the two other methods could be retained as good candidates for a real MDO implementation.

After comparisons, it appears that the best formulation for true MDO capability is the BLISS/RS. Modifications were proposed to improve the simulation models when approximated global constraints are used. Additional work should be carried out to improve the approximation features to make this method more reliable and possibly less dependent on preliminary model decisions. For example, the use of neural networks should be studied to strengthen the optimization models.

Acknowledgements

The current MDO research is supported by the New Initiative Founding from IAR/NRC.

References

[1]. Sobieszczanski-Sobieski, J and Haftka, R T. Multidisciplinary aerospace design optimization: survey of recent developments. *34th AIAA Aerospace Sciences Meeting and Exhibit*, Reno, NV, January 15-18, 1995.

[2]. Alexandrov, N M and Kodiyalam, S. Initial results of an MDO method evaluation study. *7th AIAA/USAF/NASA/ISSMO Symposium on Multidisciplinary Analysis and Optimization*, St. Louis, MO, Sept. 2-4, 1998.

[3]. Cramer, E J, Dennis, J E, Jr., Frank, P D, Lewis, R M and Shubin, G R. Problem formulation for multidisciplinary design optimizations. *SIAM Journal on Optimization*, Vol. 4, No. 4, Nov. 1994, pp. 754-776.

[4]. Sellar, R S, Batill, S M and Renaud, J E. Response surface based, concurrent subspace optimization for multidisciplinary system design. *34th AIAA Aerospace Sciences Meeting and Exhibit*, Reno, NV, Jan. 15-18 1995.

[5]. Braun, R D, and Kroo, I M. Development and application of the collaborative optimization architecture in a multidisciplinary design environment. *Multidisciplinary Design Optimization: State of the Art*, N. M. Alexandrov and M. Y, Hussaini, eds., SIAM, 1997, pp. 98-116.

[6]. Kodiyalam, S and Sobieszczanski-Sobieski, J. Bilevel integrated system synthesis with response surfaces. *AIAA Journal*, Vol. 38, No. 8, August 2000, pp. 1479-1485.

[7]. *MATLAB the Language of Technical Computing, Using MATLAB Version 6*, The MathWorks Inc., 2000.

[8]. *Optimization Toolbox for use with MATLAB, User's Guide*, The MathWorks Inc., 1999.

[9]. Alexandrov, N M and Lewis, R M. Comparative properties of collaborative optimization and other approaches to MDO, *First ASMO UK/ISSMO Conference on Engineering Design Optimization*, July 8-9, 1999.

[10]. DeMiguel, A-V and Murray, W. An Analysis of collaborative optimization methods. *8th AIAA/USAF/NASA/ISSMO Symposium on Multidisciplinary Analysis and Optimization*, Long Beach, CA, 6-8 Sept. 2000.

[11]. Alexandrov, N M and Lewis, R M. Analytical and computational aspects of collaborative optimization. *NASA Report TM-2000-210104*, April 2000.

[12]. Sobieszczanski-Sobieski, J. Sensitivity of complex, internally coupled systems. *AIAA Journal*, Vol. 28, No. 1, 1990.

[13]. Sobieszczanski-Sobieski, J. Optimization by decomposition: a step from hierarchic to non-hierarchic systems. *Second NASA/Air Force Symposium on Recent Advances in Multidisciplinary Analysis and Optimization*, Hampton, Virginia, Sept. 1988.

[14]. Sobieszczanski-Sobieski, J, Agte, J and Sandusky, R Jr. Bi-level Integrated System Synthesis (BLISS). *NASA Report TM-1998-208715*, 1998.

[15]. Sobieszczanski-Sobieski, J, Agte, J and Sandusky, R. Jr. Bi-level integrated system synthesis. *7th AIAA/USAF/NASA/ISSMO Symposium on MultidisciplinaryAnalysis and Optimization*, St. Louis, MO, Sept. 2-4, 1998.

[16]. Sobieszczanski-Sobieski, J, Emiley, M S, Agte, J and Sandusky, R Jr. Advancement of Bi-Level Integrated System Synthesis (BLISS*). 38th AIAA, Aerospace Sciences Meeting and Exhibit*, Reno, NV, Jan. 10-13, 2000.

[17]. Chan, L. Evaluation of two concurrent design approaches in multidisciplinary design optimization. *NRC Report IAR-LM-A-077*, National Research Council Canada, 2001.