Struct Multidisc Optim 19, 17–35 © Springer-Verlag 2000

A simulation-based comparison of multidisciplinary design optimization solution strategies using CASCADE

K.F. Hulme and C.L. Bloebaum

Abstract The design of multidisciplinary systems (such as aircraft, automobiles, and others) often requires an iterative cycle that includes a design initialization, a system analysis, a sensitivity analysis, and design optimization. This design cycle is standard in the field of Multidisciplinary Design Optimization (MDO) and has often been referred to in the literature as the "Multiple-Discipline-Feasible" (MDF) approach. The name stems from the fact that complete multidisciplinary feasibility is maintained in each and every design cycle. The drawback of MDF is that it can be a timely and a costly procedure. Numerous researchers have developed alternate means for posing and subsequently solving the multidisciplinary design problem. One such solution procedure has been referred to both as "Simultaneous Analysis and Design" (SAND) and "All-at-Once" (AAO), and treats the entire multidisciplinary design cycle as one large optimization problem. Another alternate solution procedure has been referred to as "Individual-Discipline-Feasible" (IDF); this procedure exhibits characteristics which lie in between the two extremes exemplified by MDF and AAO. IDF assures that each individual discipline is feasible on every design cycle, while driving the entire system (all disciplines) towards multidisciplinary feasibility. The present work will present a rigorous numerical comparison of these solution strategies over a wide variety of problem sizes and complexities. The purpose of this comparison is for the eventual development of heuristics which will govern the appropriateness of a given solution strategy for a given set of system characteristics. The multidisciplinary design test problems that are used for these comparisons are generated by a robust simulation tool called CASCADE.

Received January 12, 1999
Revised manuscript received April 28, 1999
Communicated by J. Sobieski

K.F. Hulme and C.L. Bloebaum

Multidisciplinary Optimization and Design Engineering Laboratory (MODEL), Department of Mechanical and Aerospace Engineering, State University of New York at Buffalo, 1012 Furnas Hall, Buffalo, NY 14260, USA
e-mail: clb@eng.buffalo.edu

1 Notation

AAO = All-at-Once
BV = behaviour variable (subsystem output)
CS = cost scenario
CV = coupling variable
DV = design variable
IC = inequality constraint
IDF = Individual-Discipline-Feasible
MDF = Multiple-Discipline-Feasible
OF = objective function
OV = optimization variable
SC = side constraint
SS = subsystem
SVF = system volatility factor
TS = test system

2 Background and motivation

Concurrent engineering is a systematic approach to the integrated, simultaneous design of products and their related processes. Many of the recently developed capabilities to address concurrent engineering have stemmed from the emerging area of Multidisciplinary Design Optimization, or MDO. The origins of MDO can be traced back to the early 1980's, where a linear decomposition approach (Sobieszczanski-Sobieski 1982) was used to subdivide the design of a large engineering system into a grouping of related and more manageable subsystems. However, such a decomposition often results in a grouping of subsystems which cannot be placed into a definitive top-down hierarchy. The resultant decomposition grouping is typically hybrid-hierarchic in nature as shown in the example decomposition of Fig. 1.

This inherent lack of hierarchy requires that the system analysis associated with the overall design cycle be initialized to some set of values, and iteratively converged thereafter. Subsequent to attaining a converged

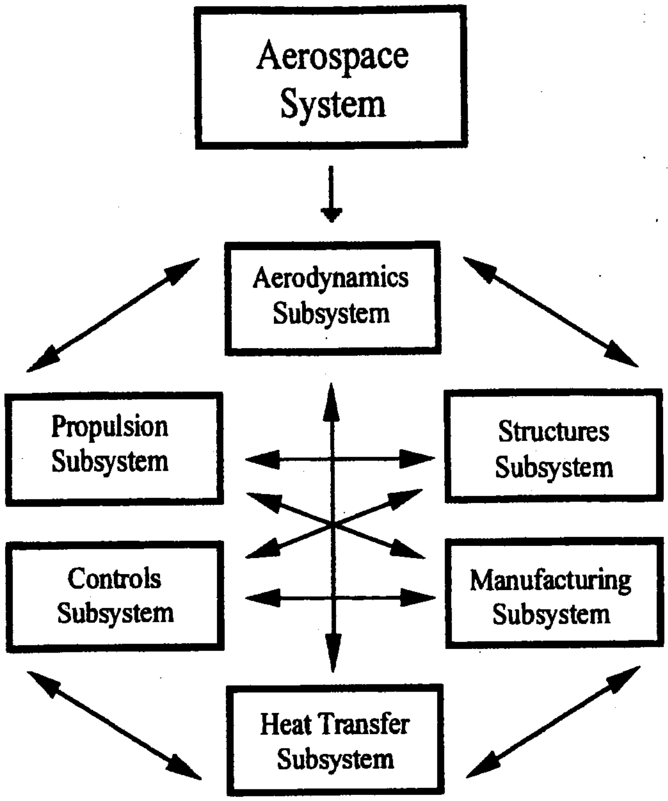

Fig. 1 Hybrid hierarchic decomposition

analysis solution, a sensitivity analysis is performed. The sensitivity analysis can be a numerical procedure such as finite differencing or an analytical procedure, namely the Global Sensitivity Equation (GSE) (Sobieszczanski-Sobieski 1990; Bloebaum *et al.* 1990) method. The sensitivity analysis is required for the optimization of the design. The optimization step itself will typically cause certain optimization variables to change, which then necessitates the reconvergence of the system analysis. Hence, the entire design cycle repeats itself until a converged solution is attained. A summary of such nonhierarchic design synthesis is illustrated in Fig. 2, and further explained in the next section.

3 Multidisciplinary solution strategies

The design cycle described above has been referred to in literature as the "Multiple-Discipline-Feasible", or MDF approach (Cramer *et al.* 1992, 1994; Balling and Sobieszczanski-Sobieski 1994). It has been demonstrated more than any other approach on nonhierarchic multidisciplinary examples. The advantages of the MDF approach include its commonality to most MDO researchers, and its optimization problem, which treats only design variables (and *not* behaviour variables) as optimization variables. The primary disadvantage of the MDF approach is that it is potentially very time and cost consuming. At each optimization iteration, complete multidisciplinary

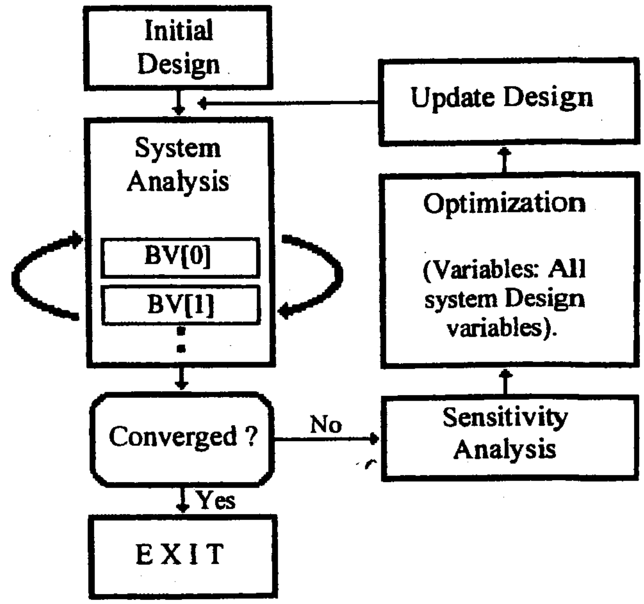

Fig. 2 Nonhierarchic design synthesis – the "Multiple-Discipline-Feasible" (MDF) strategy

feasibility is enforced. At each design cycle, a great deal of time may be inefficiently spent during the full reconvergence of the system analysis portion of a design that is still very far from its optimal solution.

More recently, researchers have focused on alternate methods for posing and solving the multidisciplinary design problem. An approach has been developed which treats the entire multidisciplinary design cycle seen in Fig. 2 as a single large optimization problem. This is accomplished by converting the system analysis equations into equality constraints, and by treating both system design variables and subsystem outputs (behaviour variables) as optimization variables. Such an approach has been referred to in the literature as both "Simultaneous Analysis and Design" (SAND) and "All-at-Once" (AAO) (Haftka 1985; Cramer *et al.* 1992, 1994; Balling and Sobieszczanski-Sobieski 1994). The primary advantage of AAO is the elimination of an iterative design cycle for attaining an optimal design through the outright elimination of costly iterative analysis evaluations. One possible disadvantage of AAO is that a much more complicated optimization problem results. More optimization variables and more equality constraints are present in the AAO formulation. These variables and equations stem from the addition of the system analysis equations to the optimization problem statement. A second disadvantage is that disciplinary feasibility is only attained at a relative or at an absolute extremum. This reduces the possibility of attaining a valid design solution if the optimizer is unsuccessful in attaining the global optimum solution. A generalized summary of the AAO strategy is seen in Fig. 3. Notice that the "residual evaluator" has replaced the iterative system analysis seen in Fig. 2. In the residual evaluator, the analysis equality constraints are posed and evaluated.

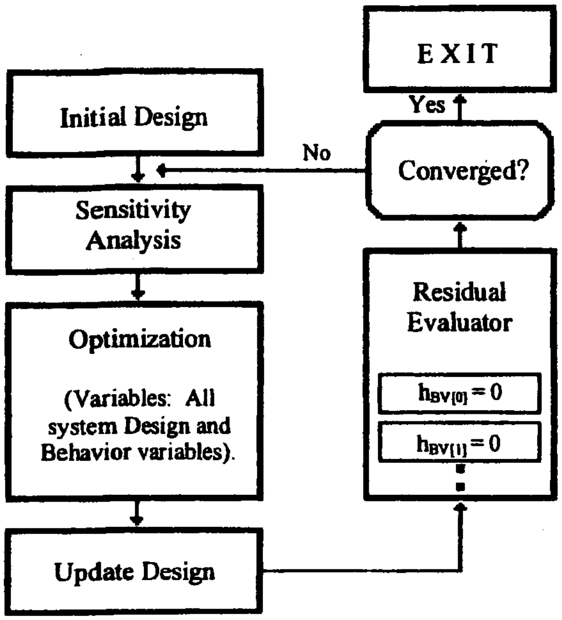

Fig. 3 The "All-at-Once" (AAO) strategy

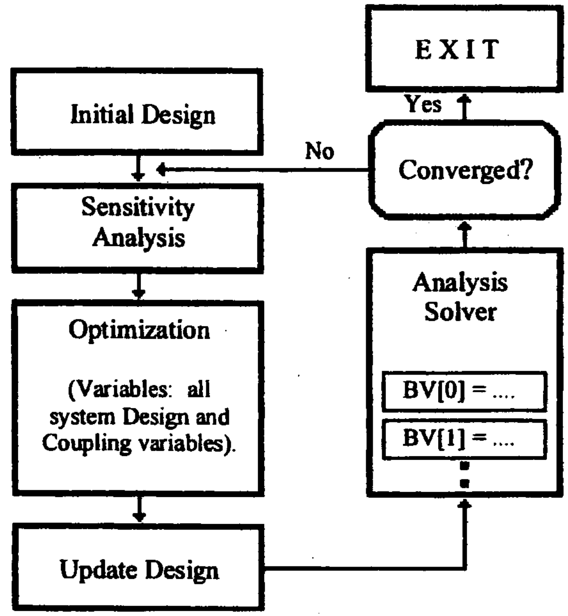

Fig. 4 The "Individual-Discipline-Feasible" (IDF) strategy

Another alternative to the classical MDF approach exhibits characteristics which lie between the extremes of MDF and AAO. Recall that MDF requires full disciplinary feasibility at each and every optimization iteration, while AAO only enforces disciplinary feasibility at the final solution (a local or global optimum), if attained. An intermediate approach has been called "Individual-Discipline-Feasible" (IDF) (Haftka *et al.* 1992; Cramer *et al.* 1992, 1994; Balling and Sobieszczanski-Sobieski 1994). With IDF, each individual discipline (or subsystem) is independently feasible at every optimization iteration. The optimizer eventually drives all of the individual disciplines towards multidisciplinary feasibility by controlling the interdisciplinary data. In this formulation, all coupling variables (behaviour variables that are required inputs to other subsystems) are promoted to being optimization variables. This takes place by temporarily substituting a replacement "surrogate" variable for each coupling variable in the optimization problem statement. Auxiliary equality constraints are added to the problem formulation to ensure that each and every behaviour variable is equal to its corresponding surrogate variable, at optimization convergence. These constraints may be thought of as "equilibrium" constraints. A generalized summary of the IDF strategy is seen in Fig. 4. Notice that the "analysis solver" has replaced the iterative system analysis seen in Fig. 2. In the analysis solver, both the single analysis solution (noniterative) and the equilibrium constraint formulation take place.

With a fundamental understanding of each of the three solution strategies, the authors' means for obtaining a wide variety of coupled multidisciplinary test systems can now be discussed.

4 Simulation of coupled systems

4.1 Motivation

A number of previous research efforts involving the comparison of the MDF, IDF, and AAO solution strategies have been extensive in theoretical detail. However, many of these research efforts have limited their implementation of the theory to simple example problems, often with easily obtainable closed-form solutions (Cramer *et al.* 1994; Balling and Wilkinson 1997). The importance of these past studies cannot be understated. In the present work, it is desired to assess the utility of these solution strategies, side-by-side, over a wide variety of system sizes and subsystem-level coupling densities. The desired problem data should have a known structure, and should have unknown (randomly generated) semantics; the global optimum solution of these test problems should be unknown, a priori. It is only after such testing is completed that heuristic rules can be developed which might help to govern the appropriateness of a given solution strategy for a given set of coupled system characteristics. To this end, the present research expands on a preliminary investigation that has been previously conducted (Hulme and Bloebaum 1998).

The rigorous testing of these solution strategies requires the use of a wide variety of stable test systems. For this task, a robust simulation tool termed CASCADE has been used. CASCADE is an acronym which stands for "Complex Application Simulator for the Creation of Analytical Design Equations". A thorough description of CASCADE can be found in past literature (Hulme and Bloebaum 1996, 1997). A brief overview will be presented here for completeness.

4.2
Simulator description and uses

CASCADE is a computer tool that generates a coupled system representation that consists of analytical equations of user-specified size. CASCADE has the capability of generating equations which represent both a coupled system analysis and an associated optimization problem. The analysis portion consists of a band of equations that attempt to represent the coupled structural nature of the subsystem outputs (behaviour variables). Each behaviour variable is a function of both independent inputs (design variables) and dependent inputs (other behaviour variables). The optimization portion consists of a system-level objective function or numerous subsystem-level objective functions, inequality constraint equations, and implicit side constraints on the optimization variables. The equations themselves are polynomial functions with positive integer exponents, and are typically highly nonlinear in nature. The design space that results from the objective function and inequality constraint equations is nonconvex, and usually clustered with a multitude of local minima. While such test systems might be extreme in the sense that they do not accurately represent the semantic behaviour of a "real world" multidisciplinary system, the authors care to view the representative behaviour of these test systems as a "worst case" scenario.

CASCADE-generated equations that are created to represent the system analysis portion of the multidisciplinary design problem can be generated at varying levels of "system volatility". System volatility serves as a meter of both the strength of the couplings and the stability of the generated system. Default CASCADE systems are generated at a system volatility of 1, which indicates a loosely coupled system of well-behaved equations that are guaranteed to remain stable within the implicit variable bounds. The user can vary the system volatility from 1 to 10, where 10 would represent a tightly-coupled, highly unstable set of equations (see Fig. 5). If a coupled set of analysis equations are thought of as an array of equality constraint contours, a low system volatility would tend to result in an optimization design space that is cluttered with many "shallow" local minima, while a large system volatility would tend to result in a design space with fewer, but "steeper" and much more drastic local minima.

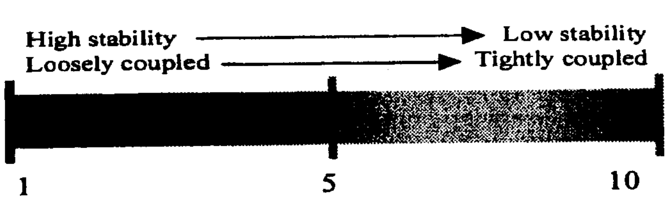

Fig. 5 System volatility factor for CASCADE

Using fixed-point iteration (which is an easily implemented but costly means for converging a coupled set of nonlinear equations), a five subsystem, twenty-five behaviour variable system with a system volatility of 1 might require 15–20 iterations to achieve convergence to six decimal places. The same size system with a system volatility of 5 might require 35–50 iterations to converge, and with a system volatility of 10, might require 100–150 iterations to converge. Note that with a system volatility of anything greater than 1, the behaviour variables are no longer implicitly guaranteed to behave within the implicit variable bounds. This becomes an issue when assigning side constraints for any behaviour variable turned optimization variable in the IDF and AAO solution strategy implementations. The next section will present an example CASCADE system, and a corresponding problem statement for each of the three solution strategies.

5
Integration of CASCADE with the multidisciplinary solution strategies

A sample CASCADE system is presented as follows. Figure 6 consists of a multidisciplinary system that has been decomposed into three inter-related subsystems. Each subsystem has its own set of independent design variables as input (the "X's"), as well as dependent behaviour variables - outputs from other subsystems - also serving as input (the "W's", "Y's", and "Z's"). The CASCADE-generated analysis equations that exhibit the coupled behaviour illustrated in Fig. 6 might appear as follows:

$$W_1 = 0.22X_W^1 + 0.05Y_2^3 - 0.46Z_1^2 + 0.73(X_W Y_2)^1 ,$$

$$W_2 = -0.96Y_2^2 + 0.56Z_1^3 - 0.03(X_W Y_2)^2 ,$$

$$W_3 = 0.36X_W^2 + 0.93(Z_1 Y_2)^2 ,$$

$$Y_1 = 0.08X_Y^3 - 0.05W_1^1 + 0.11W_3^1 - 0.09(X_Y W_1)^3 ,$$

$$Y_2 = 0.59W_3^1 + 0.41W_1^2 + 0.99(X_Y W_3)^3 ,$$

$$Z_1 = -0.43X_Z^2 - 0.88W_2^2 + 0.25(W_2 X_Z)^2 . \quad (1)$$

The CASCADE-generated optimization problem is a function of the same design and behaviour variables that are found in the analysis equations, and might appear as follows:

minimize :

$$F = 0.04X_W^2 + 0.96X_Y^3 + 0.15X_Z^1 - 0.26W_1^2 +$$

$$0.44W_2^1 + 0.57W_3^3 - 0.07Y_1^1 + 0.68Y_2^2 - 0.02Z_1^3 , \quad (2)$$

subject to:

$$g_W = -578.9 + 0.36Y_2^3 + 0.55X_W^1 + 0.09(X_W Z_1)^3 \leq 0\,,$$

$$g_Y = -226.7 + 0.26X_Y^3 + 0.51W_1^2 + 0.53(X_Y W_3)^1 \leq 0\,,$$

$$g_Z = -1095.1 + 0.33Y_2^2 + 0.47(X_Z W_2)^1 \leq 0\,,$$

$$-9999 \leq X_W, X_Y, X_Z \leq 9999\,. \qquad (3)$$

Note that CASCADE generates the inequality constraints around the converged system analysis in such a way that the initial design point is a feasible design point. The problem statements for each of the three solution strategies will be presented, corresponding to Fig. 6 and (1) through (3).

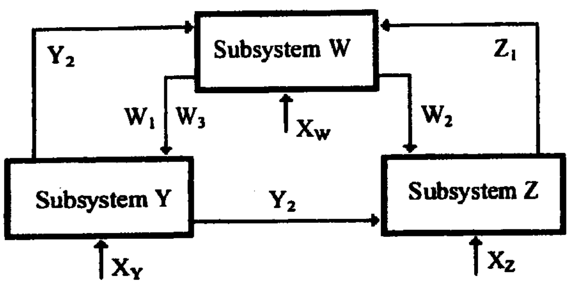

Fig. 6 Decomposed three-subsystem coupled system

5.1 MDF

Optimization variables: X_W, X_Y, and X_Z.

Analysis: Equation (1). Full iterative reconvergence every MDO cycle.

Optimization: Equations (2) and (3).

Comment. The problem is solved as posed in (1) through (3) and in a cyclic manner similar to that seen in Fig. 2.

5.2 IDF

Optimization variables: X_W, X_Y, X_Z, and surrogate variables X_{W1}, X_{W2}, X_{W3}, X_{Y2}, and X_{Z1}.

Analysis: A single noniterative "solution" of (1) on each cycle, modified as follows. Note the presence of replacement "surrogate" variables on the right-hand side of the equations

$$W_1 = 0.22X_W^1 + 0.05X_{Y2}^3 - 0.46X_{Z1}^2 + 0.73(X_W X_{Y2})^1\,,$$

$$W_2 = -0.96X_{Y2}^2 + 0.56X_{Z1}^3 - 0.03(X_W X_{Y2})^2\,,$$

$$W_3 = 0.36X_W^2 + 0.93(X_{Z1} X_{Y2})^2\,,$$

$$Y_1 = 0.08X_{Y3} - 0.05X_{W1}^1 + 0.11X_{W3}^1 - 0.09(X_Y X_{W1})^3,$$

$$Y_2 = 0.59X_{W3}^1 + 0.41X_{W1}^2 + 0.99(X_Y X_{W3})^3\,,$$

$$Z_1 = -0.43X_Z^2 - 0.88X_{W2}^2 + 0.25(X_{W2} X_Z)^2\,. \qquad (4)$$

Optimization

minimize :

$$F = 0.04X_W^2 + 0.96X_Y^3 + 0.15X_Z^1 - 0.26X_{W1}^2 + 0.44X_{W2}^1 + 0.57X_{W3}^3 - 0.07Y_1^1 + 0.68X_{Y2}^2 - 0.02X_{Z1}^3\,, \qquad (5)$$

subject to :

$$g_W = -578.9 + 0.36X_{Y2}^3 + 0.55X_{W1} + 0.09(X_W X_{Z1})^3 \leq 0\,,$$

$$g_Y = -226.7 + 0.26X_Y^3 + 0.51X_{W1}^2 + 0.53(X_Y X_{W3})^1 \leq 0\,,$$

$$g_Z = -1095.1 + 0.33X_{Y2}^2 + 0.47(X_Z X_{W2})^1 \leq 0\,,$$

$$0 = X_{W1} - W_1\,,\quad 0 = X_{W2} - W_2\,,\quad 0 = X_{W3} - W_3\,,$$

$$0 = X_{Y2} - Y_2\,,\quad 0 = X_{Z1} - Z_1\,,$$

$$-9999 \leq X_W, X_Y, X_Z, X_{W1}, X_{W2}, X_{W3}, X_{Y2}, X_{Z1} \leq 9999\,. \qquad (6)$$

Comment. Note that Y_1 is not an optimization variable, since it is not required as input by any subsystem.

5.3 AAO

Optimization variables: X_W, X_Y, X_Z, W_1, W_2, W_3, Y_1, Y_2, and Z_1.

Analysis: None. ("Analysis" is included in the optimization problem.)

Optimization

minimize :

$$F = 0.04X_W^2 + 0.96X_Y^3 + 0.15X_Z^1 - 0.26W_1^2 + 0.44W_2^1 + 0.57W_3^3 - 0.07Y_1^1 + 0.68Y_2^2 - 0.02Z_1^3\,, \qquad (7)$$

subject to :

$$g_W = -578.9 + 0.36Y_2^3 + 0.55X_W^1 + 0.09(X_W Z_1)^3 \le 0\,,$$

$$g_Y = -226.7 + 0.26X_Y^3 + 0.51W_1^2 + 0.53(X_Y W_3)^1 \le 0\,,$$

$$g_Z = -1095.1 + 0.15Y_1^1 + 0.33Y_2^2 + 0.47(X_Z W_2)^1 \le 0\,,$$

$$0 = 0.22X_W^1 + 0.05Y_2^3 - 0.46Z_1^2 + 0.73(X_W Y_2)^1 - W_1\,,$$

$$0 = -0.96Y_2^2 + 0.56Z_1^3 - 0.03(X_W Y_2)^2 - W_2\,,$$

$$0 = 0.36X_W^2 + 0.93(Z_1 Y_2)^2 - W_3\,,$$

$$0 = 0.08X_Y^3 - 0.05W_1^1 + 0.11W_3^1 - 0.09(X_Y W_1)^3 - Y_1\,,$$

$$0 = 0.59W_3^1 + 0.41W_1^2 + 0.99(X_Y W_3)^3 - Y_2\,,$$

$$0 = -0.43X_Z^2 - 0.88W_2^2 + 0.25(W_2 X_Z)^2 - Z_1\,,$$

$$-9999 \le X_W, X_Y, X_Z, W_1, W_2, W_3, Y_1, Y_2, Z_1 \le 9999\,. \qquad (8)$$

Comment. All design and behaviour variables are controlled by the optimizer.

With a general understanding of the three different means for posing the MDO test systems that are generated by CASCADE, the results of several numerical comparisons can be presented. Prior to doing so, a brief description of both the three simulation types and general optimization details is offered.

6 Simulation description — Test systems and optimization details

CASCADE has been used to generate a multitude of simulations of coupled multidisciplinary systems. These test systems vary both in size and in coupling complexity. Section 7.1 presents a comparison of five test systems which were tested "extensively". The idea here was to gain strong preliminary insight into the dynamic of each of the solution strategies by spending a substantial amount of effort testing a small number of test systems. For each solution strategy and for each test problem, more than 100 trial executions were performed, where varying optimizer settings were attempted. Section 7.2 presents a comparison of test systems which were tested "moderately". The idea here was to test a larger number of test systems in a relatively short period of time. Here, forty-five systems were tested in total; five test system sizes were used, with a total of nine different "instances" (same structure, different semantics) of each system generated for each system size. Well-trusted "default" optimizer settings were used, the insight to which was gained during the Section 7.1 testing. Finally, Section 7.3 presents a comparison of three "moderately" tested systems, while specifically taking into consideration the simulated cost associated with the execution of each of the solution strategies. The idea here was to illustrate that in addition to the final objective function value attained, the system manager must also take into consideration the amount of resources required to achieve the final solution for the particular solution strategy that has been chosen.

The ANSI-C translated version of Automated Design Synthesis (ADS) (Vanderplaats 1985) has been used as the optimization software for these MDO test systems. The strategy-optimizer combination that has been used for the acquisition of all of the simulation results is *Sequential Linear Programming - Method of Feasible Directions*. Internal finite difference methods have been used to attain gradient information, within ADS. All trial executions were performed on a *SUN Ultra 1 Creator 3D* workstation, under comparable network conditions. Fixed-point iteration was used as the iterative technique for converging the analysis equations, when the MDF solution strategy was implemented.

7 Results

7.1 Preliminary testing

For the first set of trial executions, the authors tested each of the solution strategies in great depth for each of the five test problems. Many hours were spent altering optimizer settings and performing a multitude of trial executions. Testing was halted only after an enormous number of trial attempts were made, and after the authors felt confident that no more substantial improvement could be gained with a "reasonable" amount of further effort. Hence, the data presented should be a good indicator of the "best" attainable result for each of the solution strategies. Primary characteristics of each of the five test systems are summarized in Table 1. Recall that the number of *coupling variables* for each test system corresponds to the number of behaviour variables of a system which are required as input by at least one other subsystem.

Test system 1 has three subsystems (W, Y, and Z), which have 2, 1, and 3 behaviour variables per subsystem, 3, 2, and 1 design variables per subsystem, and 1, 3, and 2 inequality constraints per subsystem, respectively. The initial value of the objective function is -678.71. Figure 7

Table 1 Summary of the five test systems (preliminary testing)

Test System	No. SS's	No. BV's	No. DV's	No. IC's	No. CV's	Initial OF
1	3	6	6	6	5	−678.71
2	5	9	11	10	8	−43.91
3	10	20	40	20	12	1622.74
4	15	45	45	90	43	−820.18
5	20	100	40	3	92	197.40

provides a detailed illustration of the coupling structure of the first test system. Figure 8 compares the test system 1 objective function histories for all three solution strategies. All solution strategies achieve approximately the same optimal design point, with the MDF strategy attaining the lowest objective function value of −1711.79. Table 2 summarizes the "best" results attained (where "best" implies lowest objective function) for all five test systems, for each of the three solution strategies.

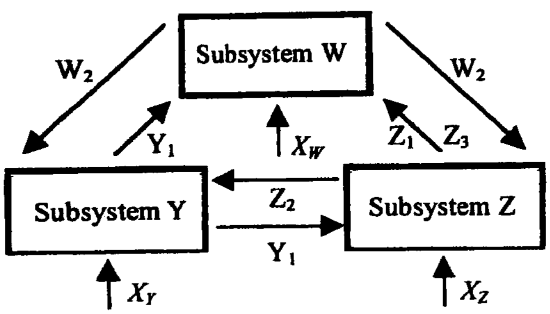

Fig. 7 Schematic of test system 1

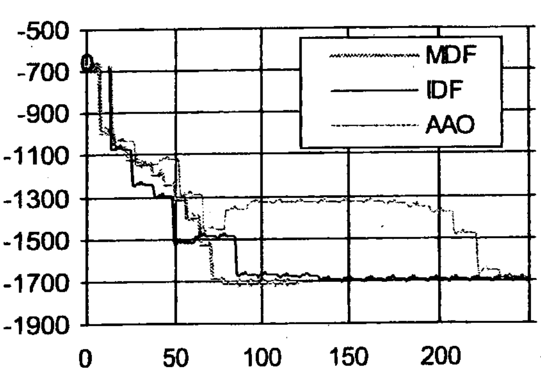

Fig. 8 Test system 1 – Objective function value vs. evaluation number

Test system 2 has five subsystems (U, V, W, Y, and Z), which have 1, 2, 1, 3, and 2 behaviour variables per subsystem, 3, 2, 1, 2, and 3 design variables per subsystem, and 4, 0, 1, 2, and 3 inequality constraints per subsystem, respectively. The initial value of the objective function is −43.91. Figure 9 provides a detailed illustration of the coupling nature of the second test system, and Figure 10 compares the test system 2 objective function histories for all three solution strategies. Again, all solution strategies achieve approximately the same optimal design point. Here again, the MDF strategy attains the lowest objective function value of −3150.08. Refer to Table 2 for tabular results.

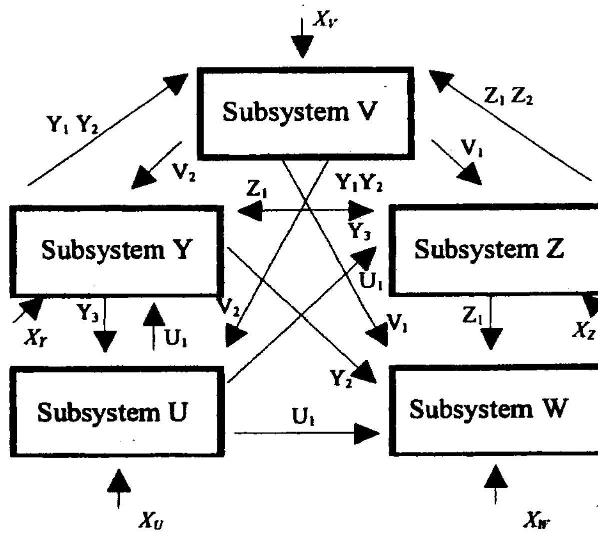

Fig. 9 Schematic of test system 2

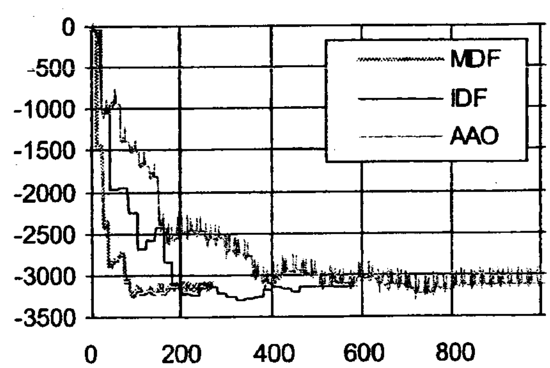

Fig. 10 Test system 2 – Objective function value vs. evaluation number

Test system 3 has ten subsystems, a total of twenty behaviour variables, forty design variables, and twenty inequality constraints. (For the final three test systems, detailed coupling illustrations and objective function histories are omitted for brevity.) The initial value of the objective function is 1622.74. Once again, all solution strategies achieve approximately the same optimal design point. The MDF strategy attains the lowest objective function value of −11859.7. Refer to Table 2.

Table 2 Result summary for test systems 1–5

TS	Sol. strat.	Total OV's	Final OF	Active SC's	Active IC's	Run time (sec)
1	MDF	6	−1711.79	3	1	0.486
	IDF	11	−1706.45	3	1	1.916
	AAO	12	−1705.15	3	1	2.029
2	MDF	11	−3150.08	4	1	1.094
	IDF	19	−3144.08	4	1	7.245
	AAO	20	−3124.61	4	1	9.146
3	MDF	40	−11859.7	15	4	22.927
	IDF	52	−11407.5	13	6	27.023
	AAO	60	−11316.8	13	6	64.201
4	MDF	45	−10118.8	13	22	57.854
	IDF	88	−8882.84	11	10	307.943
	AAO	90	−8483.45	11	12	260.995
5	MDF	40	−12012.6	26	2	98.179
	IDF	132	−10253.4	13	2	1892.150
	AAO	140	−10980.8	14	2	3098.899

Test system 4 has fifteen subsystems, a total of forty-five behaviour variables, forty-five design variables, and ninety inequality constraints. The initial value of the objective function is −820.18. Here, all three solution strategies do not arrive at "equivalent" solutions. The MDF strategy attains the lowest objective function value of −10118.8, which is approximately 10% lower than the next best solution attained, by the IDF strategy. Refer to Table 2.

Finally, test system 5 has twenty subsystems, a total of one-hundred behaviour variables, forty design variables, and only three inequality constraints. The initial value of the objective function is 197.40. Here again, there is a distinct difference between the final solutions reached by each of the three solution strategies. MDF again achieves the lowest objective function value of −12012.6, considerably lower than the next best solution attained, by the AAO strategy. Refer to Table 2.

A final plot presented in this subsection of results is shown in Fig. 11, which is a plot of total execution time vs. test system for all three solution strategies. Notice that the execution times are plotted logarithmically along the y-axis (i.e. "1" $= 10^1$ seconds, "3" $= 10^3$ seconds, etc.). The MDF strategy achieves its final solution the most quickly for all five test systems, followed by the IDF strategy (for four of the five test systems), followed by the AAO strategy, which requires the most time to attain its final solution (for all but one of the test systems). Note also that execution times increase dramatically with system size, as does the rate of the increase in execution time. Test system 1, a relatively small test system which has six design and six behaviour variables, requires no more than 3 seconds to solve for all three strategies. Test system 3, a moderately sized test system with twenty behaviour variables and forty design variables, requires approximately 23 seconds to solve with the MDF strategy, and almost 65 seconds to solve with the AAO strategy. Conversely, test system 5 is a relatively large test system which has one-hundred behaviour variables and forty design variables. This test system requires approximately 100 seconds to solve with the MDF strategy, and almost 3100 seconds to solve with the AAO strategy.

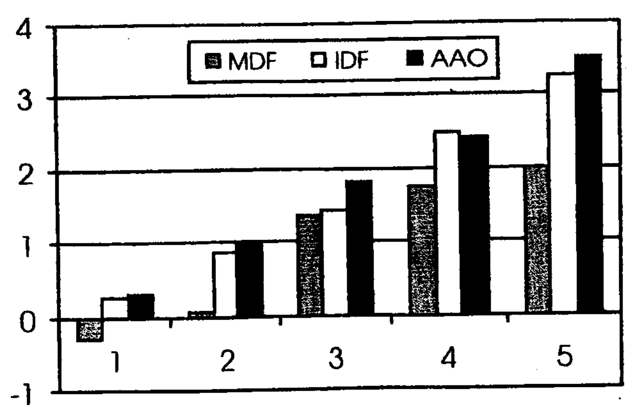

Fig. 11 Execution time ($\log_{10}$ seconds) vs. test system

7.2 Verification testing

For the second set of trial executions, the authors took a different approach. Having previously gained insight to the behaviour of each of the solution strategies under a wide range of optimizer settings, the number of trial executions for each test system and for each solution strategy was to now be limited. The limit for each test system instance was set to be the number of attempts required to attain five improved solutions (over the initial feasible solution), or ten total trial executions; whichever was arrived at first. In addition, at least one improved solution was required. This allowed the authors to gain a large cross-section of data rather quickly, while using a reliable assignment of optimizer settings. These reliable optimizer settings were held constant; the only optimizer parameters that were allowed to change between trial executions were the relative move limit parameter (for all solution strategies), the binary variable-scaling switch (for all solution strategies), and the penalty parameter for equality constraints (for AAO and IDF only). Hence, this stage of testing provides the system manager with an indication of which solution strategy can be relied upon to achieve a good solution quickly, having made only minor modifications to a "standard" and trustworthy set of optimizer settings.

Forty-five systems were tested in all. Five system sizes were used, and nine distinct instances of each system were created and tested for each system size. The nine instances represent three distinct system structures at each of three levels of "system volatility"; see Section 4.2. Primary characteristics of each of the five test system sizes are summarized in Table 3.

Table 3 Summary of the five test system sizes (verification testing)

System size	No. of SS's	No. of BV's	No. of DV's	No. of IC's	No. of CV's
1	2	6	6	2	5
2	3	9	9	6	8
3	5	15	15	15	13
4	10	40	40	20	35–36
5	12	48	48	24	39–42

System size 1 has two subsystems which have three behaviour variables per subsystem, three design variables per subsystem, and one inequality constraint per subsystem, respectively. Again, there were nine instances of system size 1 tested; three each at system volatility of 1, 3, and 5, respectively. Tabular result comparisons for all simulations are listed in Table 4. The MDF strategy achieves the lowest final objective function in five of the nine system size 1 instances, those being instances 2, 5, 7, 8, and 9. The IDF strategy achieves the lowest final objective function in system size 1 instances 3 and 6, and the AAO strategy achieves the lowest final objective function in system size 1 instances 1 and 4.

Figure 12 presents a slightly different interpretation of these same results. For each of the nine system instances, the objective function improvements for the three solution strategies are normalized against the solution strategy which attained the lowest final objective function for that given system instance. For example, for instance 1 of system size 1, AAO achieved the greatest improvement in the objective function, decreasing from a starting value of −1187.97, to a final value of −2038.7 (see Table 4); marking a total improvement of 850.74. The MDF and IDF strategies saw decreases in the objective function to −1870.6 and −1966.4 (see Table 4), respectively, corresponding to objective function improvements of 682.69 and 778.45, respectively. Hence, normalized improvements for MDF, IDF, and AAO for instance 1 of system size 1 are: 0.802, 0.915, and 1.0, respectively. These numbers are plotted as the first of nine sets of bars in Fig. 12. Note that the labels on the independent axis of Fig. 12 (and subsequent plots that are analogous to it) are sequences of two numbers – the first is the instance number, and the second is the system volatility factor, displayed in parentheses. For instance 2 of system size 1, the improvements for each of the three solution strategies are normalized against the MDF improvement, since MDF showed the greatest improvement in the objective function, for that particular instance of system size 1.

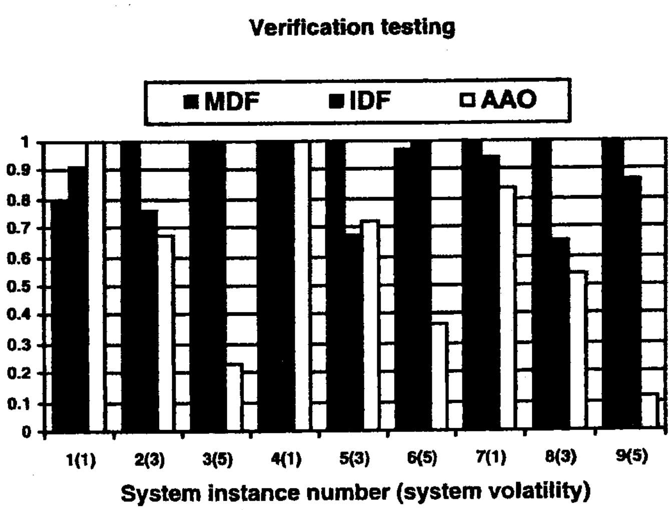

Fig. 12 Normalized improvement for system size 1

System size 2 has three subsystems which have three behaviour variables per subsystem, three design variables per subsystem, and two inequality constraints per subsystem, respectively. Again, there were nine instances of system size 2 tested; three each at system volatility of 1, 3, and 5, respectively. Result comparisons are listed in Table 4. The MDF strategy achieves the lowest final objective function in seven of the nine system size 2 instances, those being instances 2, 3, 5, 6, 7, 8, and 9. The IDF strategy achieves the lowest final objective function in system size 2 instance 4, and the AAO strategy achieves the lowest final objective function in system size 2 instance 1. Figure 13 presents a "normalized" com-

Table 4 Result summary for system sizes 1–5

System size	SVF	Inst no.	Initial OF	Final OF (MDF)	Final OF (IDF)	Final OF (AAO)
1	1	1	−1187.97	−1870.6	−1966.4	−2038.7
		4	−761.44	−2106.0	−2105.7	−2106.1
		7	−689.7	−2076.2	−2003.5	−1880.3
	3	2	1428.91	−2102.7	−1776.2	−1668.7
		3	826.39	−2518.4	−1438.1	−1601.6
		8	639.14	−2417.7	−1747.1	−1535.0
	5	3	−861.91	−2627.1	−2627.4	−1137.2
		5	−504.22	−5235.0	−5427.3	−1566.4
		9	−340.64	−22031.	−19174.	−2848.7
2	1	1	30.84	−749.8	−778.08	−975.43
		4	−63.	−746.81	−970.37	−954.2
		7	776.14	−3229.2	−2412.0	−2419.5
	3	2	−327.23	−5250.7	−3180.2	−1065.4
		3	328.37	−2258.0	−2050.3	−1519.5
		8	−1378.1	−4748.2	−1679.8	−1555.6
	5	3	−606.01	−2344.5	−1681.0	−1205.7
		5	1517.62	−2940.0	−2333.8	−1959.8
		9	968.32	−5966.8	−1442.8	−1718.2
3	1	1	259.94	−3298.7	−1303.5	−1425.36
		4	20.44	−3433.79	−2353.44	−1754.06
		7	−623.77	−4208.31	−1424.02	−1824.62
	3	2	−714.6	−6095.35	−5425.83	−5261.48
		3	173.27	−4033.66	−3368.	−2787.12
		8	−1964.36	−7295.89	−7413.61	−6598.22
	5	3	565.06	−3348.61	−2880.12	−2730.17
		5	−909.6	−3877.41	−2162.83	−2075.66
		9	468.95	−4013.51	−2612.87	−2204.04
4	1	1	−666.89	−9581.46	−5623.66	−4982.23
		4	159.27	−13143.95	−2526.37	−1402.99
		7	2092.89	−13752.55	125.41	106.21
	3	2	1091.82	−6775.22	−3615.68	−4474.38
		3	4796.72	−9372.94	731.76	830.41
		8	−4079.22	−14793.08	−5454.39	−4985.31
	5	3	−1014.9	−9540.44	−3410.96	−3798.6
		5	443.29	−13373.53	−2746.87	−2729.25
		9	−492.71	−11101.14	−2463.47	−1953.93
5	1	1	774.8	−21308.78	−9050.62	−7824.56
		4	−4193.6	−18576.07	−6938.9	−7123.5
		7	273.5	−13735.75	−1369.76	−949.01
	3	2	108.19	−14632.72	−3378.39	−4882.27
		3	2128.57	−5036.72	−4.46	75.11
		8	4702.5	−11953.16	531.84	1763.89
	5	3	−1702.66	−14513.31	−3008.24	−3282.05
		5	2357.14	−14262.63	1.53	−2150.55
		9	2292.28	−11847.47	−312.56	92.83

parison of the nine instances of system size 2, as explained in the previous paragraph.

System size 3 has five subsystems which have three behaviour variables per subsystem, three design variables per subsystem, and three inequality constraints per subsystem, respectively. Again, there were nine instances of system size 3 tested; three each at system volatility of 1, 3, and 5, respectively. Result comparisons are listed in Table 4. The MDF strategy achieves the lowest final objective function in eight of the nine system size 3 instances, the exception being instance 6, where the IDF strategy slightly outperforms the MDF strategy. Fig-

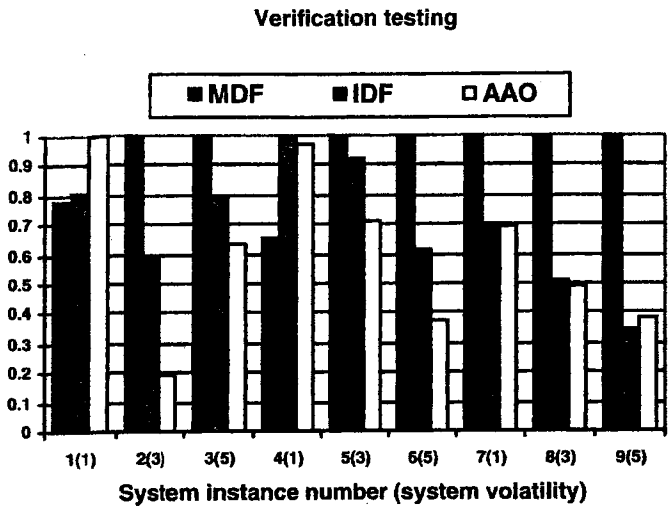

Fig. 13 Normalized improvement for system size 2

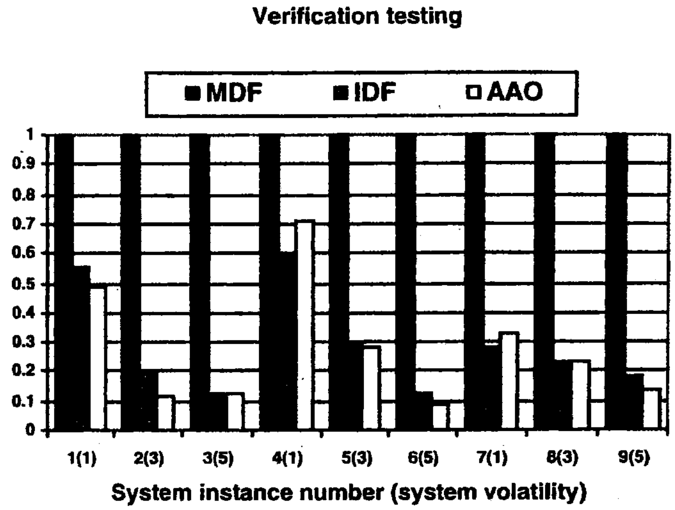

Fig. 15 Normalized improvement for system size 4

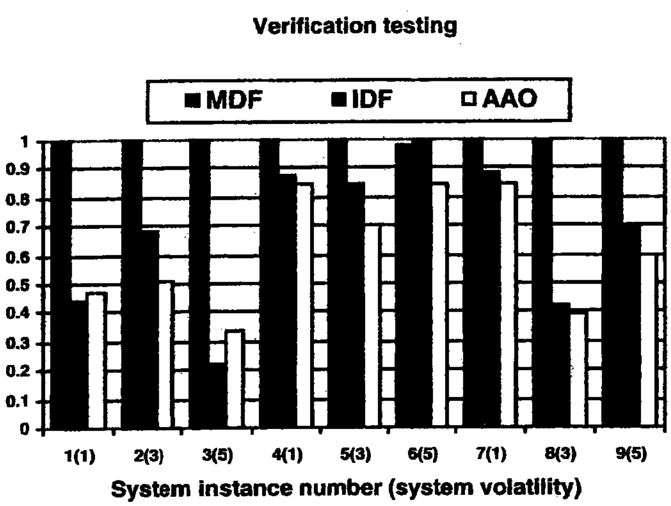

Fig. 14 Normalized improvement for system size 3

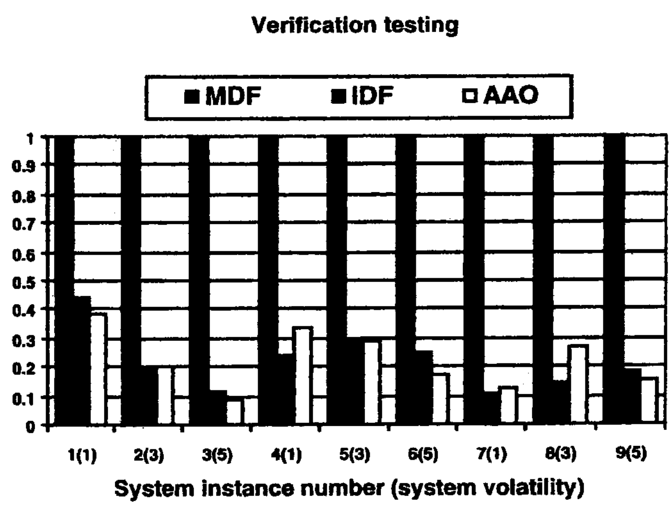

Fig. 16 Normalized improvement for system size 5

ure 14 presents a "normalized" comparison of the nine instances of system size 3.

System size 4 has ten subsystems which have four behaviour variables per subsystem, four design variables per subsystem, and two inequality constraints per subsystem, respectively. System size 5 has twelve subsystems which have four behaviour variables per subsystem, four design variables per subsystem, and two inequality constraints per subsystem, respectively. Once again, nine instances of both system size 4 and system size 5 were tested; three each at system volatility of 1, 3, and 5, respectively. Result comparisons are listed in Table 4. The MDF strategy achieves the lowest final objective function for all instances of system sizes 4 and 5. Figures 15 and 16 present a "normalized" comparison of the nine instances of system sizes 4 and 5, respectively.

A final set of tabular results for this subsection of results begins with Table 5. Table 5 lists the numerical average of the normalized improvements for all nine instances of each system size. For all system sizes, MDF has the highest average normalized improvement, which is greater than 0.93 for all system sizes. (Realize that an "average" score of 1.0 for a given system size and solution strategy would indicate that the given solution strategy achieved the greatest improvement for all nine system instances). The IDF strategy has the second highest average normalized improvement for all but the fifth system size, where the AAO strategy slightly outperformed the IDF strategy. Table 6 decomposes the results of Table 5, by separately listing the average normalized improvements for each of the three system volatility factors (1,3,5), respectively.

Table 5 Overall average normalized improvement

System size	MDF	IDF	AAO
1	0.974511	0.867397	0.609992
2	0.936463	0.696686	0.606587
3	0.997599	0.673029	0.616955
4	1	0.288155	0.276975
5	1	0.218398	0.225778

Table 6 Average normalized improvement for each system volatility level

SVF	System size	MDF	IDF	AAO
1	1	0.934150	0.951706	0.946229
	2	0.809389	0.83078	0.888729
	3	1	0.731733	0.72019
	4	1	0.478485	0.506041
	5	1	0.261119	0.283744
3	1	1	0.694386	0.647022
	2	1	0.676974	0.466061
	3	1	0.650427	0.536771
	4	1	0.239882	0.208988
	5	1	0.210100	0.253840
5	1	0.989384	0.95609	0.236723
	2	1	0.58230	0.464972
	3	0.992799	0.636926	0.593898
	4	1	0.146098	0.115897
	5	1	0.183974	0.139749

7.3 Simulated cost-based testing

7.3.1 Simulation overview

The primary theme of this final subsection of results is that of "trade-off". The first two subsections of results have focused solely on the ability of each of the solution strategies to achieve design improvement through a decrease in the system-level objective function. However, design improvement is not the only characteristic that must be considered by the design manager. Inevitably, the implementation of each of these solution strategies will have an associated cost. This cost might stem from computational resources expended (i.e. CPU time), from man hours invested in carrying through the solution process of the given system as posed, or from any of a number of other sources. In this research effort, separate simulated and generic cost amounts have been assigned to each analysis evaluation, each objective function evaluation, and each constraint function evaluation. The goal with this study was to arrive at a simulated total cost amount associated with implementing each of the solution strategies. These cost amounts are then compared side by side. The design manager can weigh the gains obtained in the optimized design versus the cost incurred in arriving there, for each of the solution strategies. Three systems were tested for this final phase of result acquisition; the characteristics of these systems are summarized in Table 7.

Table 7 Summary of the three test systems (cost-based testing)

Test sys.	No. SS's	No. BV's	No. DV's	No. IC's	No. CV's	Initial OF
1	10	10	10	20	9	−76.37
2	10	20	30	10	18	−577.26
3	10	50	50	5	47	−1436.3

7.3.2 Simulation details

In this simulation the overall cost in implementing a solution strategy is a function of three distinct components – analysis cost, objective function cost, and constraint function cost. The objective function cost is analogous for all three solution strategies; the details of the other two components differ slightly for each solution strategy. For the MDF strategy, the analysis cost is the cost of an analysis evaluation for each behaviour variable multiplied by the number of iterations required to converge the system of coupled analysis equations, summated every design cycle. The constraint function cost is the sum of the cost of each of the system inequality constraints, summated every design cycle. For the IDF strategy, the analysis cost is the cost of an analysis evaluation for each behaviour variable, summated every design cycle. The constraint function cost is the sum of the cost of each of the system inequality constraints, plus the sum of the cost for each of the equilibrium constraints corresponding to each coupling variable, summated every design cycle. For the AAO strategy, there is no analysis evaluation cost. The constraint function cost is the sum of the cost of each of the system inequality constraints, plus the sum of the cost of each analysis equation (where here each analysis equation is posed as an equality constraint), summated every design cycle.

Three "cost scenarios" are used in the presentation of the results. These scenarios are summarized in Table 8. Cost scenario 1 is such that the cost of an analysis evaluation is approximately equivalent to the cost of both an objective function evaluation and an inequality constraint function evaluation. Here, the cost of all three function evaluations are randomly chosen to be between 0 and 100 units. The authors speculate that this is not a realistic cost scenario. The cost of an analysis evaluation (which, in a real system might stem from the result of a costly FEM matrix inversion, for example) could be far greater than the evaluation cost of the optimization functions, linear or nonlinear. In response to this line of thinking are more realistic cost scenarios 2 and 3. Cost scenario 2 represents a "semi-costly" analysis, where the costs of analysis evaluations are randomly chosen to be between 0 and 500 units, and objective and inequality constraint function costs are again randomly chosen to be between 0 and 100 units. Hence, on average, the analysis/optimization evaluation cost ratio is equal to 5.0 for cost scenario 2. Cost scenario 3 represents a "costly" analysis, where the costs of analysis evaluations are randomly chosen to be between 500 and 1000 units, and objective and inequality constraint function costs are again randomly chosen to be between 0 and 100 units. Hence, on average, the analysis/optimization evaluation cost ratio is equal to 15.0 for cost scenario 3. (Note: relative to all other cost quantities, the evaluation cost of the equilibrium constraints IDF strategy could be considered negligible. However, in this simulation, the cost of an equilibrium constraint function evaluation has been set to be equal to 10% of the evaluation cost of its associated analysis equation).

Table 8 Summary of the three cost scenarios

Cost scenario	Analysis cost range	Optimization cost range	Analysis/ Optimization cost ratio
1	(0,100)	(0,100)	1.0
2	(0,500)	(0,100)	5.0
3	(500,1000)	(0,100)	15.0

7.3.3 Simulation results

Table 9 presents a comparison of the three solution strategies for all three test systems and cost scenarios. Consider first test system 1, cost scenario 1. Listed for each solution strategy are five quantities: objective function improvement, analysis evaluation cost, objective function evaluation cost, constraint function evaluation cost, and total evaluation cost. Figure 17 presents a plot of this very same data. Note however, that each of these five quantities are plotted normalized, and are normalized separately. (i.e. the three objective function improvements are normalized against each other, the three analysis costs are normalized against each other, the three objective function costs are normalized against each other, etc.) The general trends of this first system are as follows: MDF achieves the greatest objective function improvement, but has a huge analysis cost and by far the highest overall cost. AAO attains substantial improvement in the objective function, has no iterative analysis cost whatsoever, but has a large constraint cost (recall that the analysis equations are now posed as equality constraints), and has in this case, the largest objective function cost. IDF attains substantial improvement (albeit the lowest of the three strategies) at a low analysis cost, and at the lowest objective function, constraint, and total cost, respectively. Table 9 and Figs. 18 and 19 present analogous results for the more realistic cost scenarios 2 and 3, respectively, for test system 1. In comparing Figs. 18 and 19 to Fig. 17, note that the objective function improvements are the same, the analysis and objective function costs are higher, but proportionately so, the constraint cost for IDF grows to be larger than that for MDF, and the proportion by which the total cost for MDF is larger than that for AAO and IDF becomes larger. Table 9 presents analogous results for the second and third test systems, respectively.

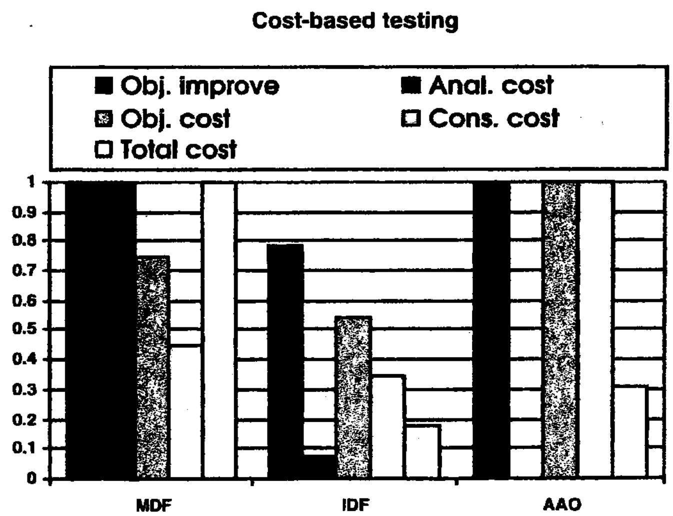

Fig. 17 Normalized comparisons for test system 1, cost scenario 1

Figures 20 and 21 are normalized plots of the cost scenario 3 data for the second and third test systems, respectively. The second test system is typified by the following: the objective function improvement of MDF doubles that of both IDF and AAO; a huge MDF analysis cost, ten times larger than that of IDF; comparable objective function costs, with that of MDF being the largest, followed by IDF and then by AAO; the constraint cost is largest for AAO, followed by IDF for the second and third cost scenarios (as with the first test system); MDF has the highest overall cost by an increasing proportion (as cost is increased from cost scenario 1 to cost scenarios 2 and 3), followed by IDF, and followed by AAO, which is the least costly solution strategy for the second test system. The third test system is typified by the following: the objective function improvement of MDF that nearly quintuples that of both IDF and AAO; a huge MDF analysis cost, ten times larger than that of IDF; comparable (and comparatively miniscule) objective function

Table 9 Cost scenarios 1–3 for test systems 1–3

TS	CS	Sol strat.	OF imprv.	Anal. cost	OF cost	Cons. cost	Total cost
1	1	MDF	2337.	2.16E6	12247	3.40E5	2.51E6
		IDF	1831.	1.69E5	8917	2.61E5	4.40E5
		AAO	2324.	0	16354	7.64E5	7.81E5
	2	MDF	2337.	1.08E7	12247	3.40E5	1.12E7
		IDF	1831.	8.54E5	8917	3.21E5	1.18E6
		AAO	2324.	0	16354	2.01E6	2.03E6
	3	MDF	2337.	2.62E7	12247	3.40E5	2.65E7
		IDF	1831.	2.06E6	8917	4.29E5	2.50E6
		AAO	2324.	0	16354	4.22E6	4.23E6
2	1	MDF	7276.	7.20E6	37710	4.10E5	7.65E6
		IDF	3349.	7.66E5	30915	3.98E5	1.19E6
		AAO	3329.	0	27585	9.83E5	1.01E6
	2	MDF	7276.	3.62E7	37710	4.10E5	3.67E7
		IDF	3349.	3.85E6	30915	6.70E5	4.55E6
		AAO	3329.	0	27585	3.73E6	3.76E6
	3	MDF	7276.	1.00E8	37710	4.10E5	1.01E8
		IDF	3349.	1.07E7	30915	1.28E6	1.20E7
		AAO	3329.	0	27585	9.86E6	9.88E6
3	1	MDF	14581.	2.45E7	3828	2.00E5	2.47E7
		IDF	3302.	2.19E6	2649	3.31E5	2.52E6
		AAO	3106.	0	6063	5.33E6	5.33E6
	2	MDF	14581.	1.24E8	3828	2.00E5	1.24E8
		IDF	3302.	1.10E7	2649	1.18E6	1.22E7
		AAO	3106.	0	6063	2.56E7	2.56E7
	3	MDF	14581.	3.71E8	3828	2.00E5	3.71E8
		IDF	3302.	3.31E7	2649	3.25E6	3.64E7
		AAO	3106.	0	6063	7.61E7	7.61E7

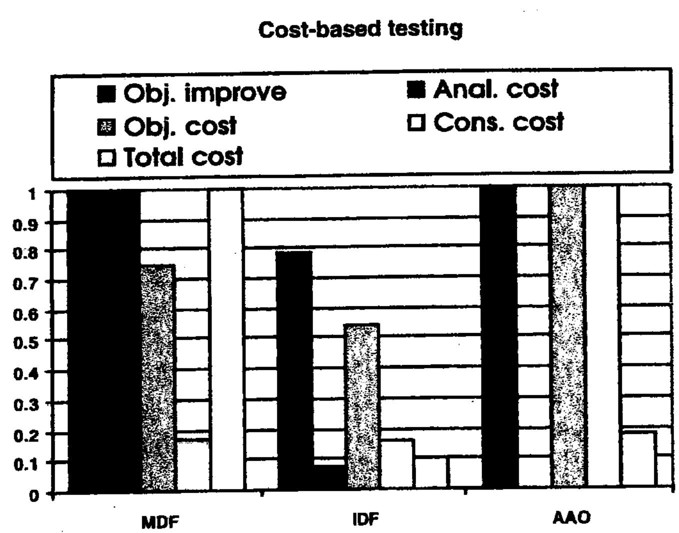

Fig. 18 Normalized comparisons for test system 1, cost scenario 2

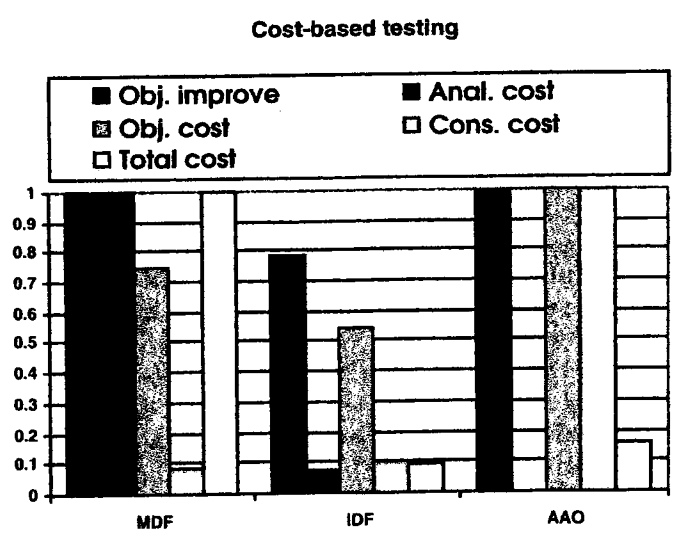

Fig. 19 Normalized comparisons for test system 1, cost scenario 3

costs, with that of MDF being the largest, followed by AAO and then by IDF; the constraint cost is largest for AAO, followed by IDF for all three cost scenarios; MDF has the highest overall cost by an increasing proportion (as cost is increased from cost scenario 1 to cost scenarios 2 and 3), followed by AAO, and followed by IDF, which is the least costly solution strategy for the third test system.

The next section will take a closer look at all three subsections of results, and will consequently make some heuristic observations regarding the utility of each of the solution strategies.

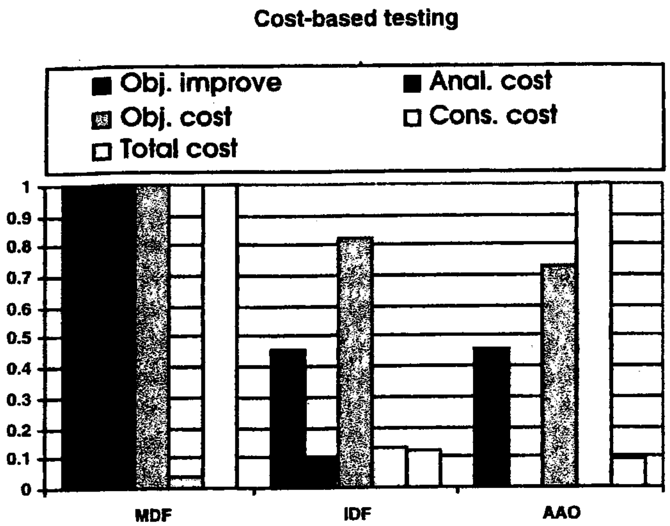

Fig. 20 Normalized comparisons for test system 2, cost scenario 3

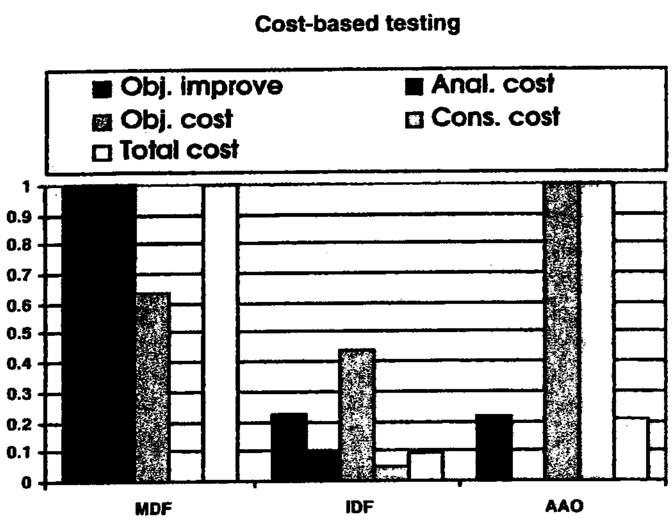

Fig. 21 Normalized comparisons for test system 3, cost scenario 3

8 Discussion

8.1 General observations

The first major point to be made regarding the results is that by having multiple solution strategies to pose and solve the same coupled multidisciplinary design problem, a design manager will arrive at three different problem statements, each with its own "problem dynamic". The solution path for each problem will likely be totally unique, and highly dependent on the initial design point. Hence, having three separate means for posing and solving an MDO problem can only be advantageous to a design manager. This was clearly found to be the case in the present work. The authors would typically perform several trial executions on a given test problem using a given solution strategy, and attain what appeared to be the "global" optimum solution. Thereafter, numerous trial executions of the same test problem with a different solution strategy would often result in an improved solution, hence revealing the initial "global" optimum solution as being merely an improved local optimum solution.

The "problem dynamic" that is mentioned in the previous paragraph results from the nature of each solution strategy; the means with which the multidisciplinary design problem is posed for a given strategy. Problems which contain a large system analysis portion (i.e. problems which have a large number of behaviour variables) will result in a large iterative system analysis for the MDF strategy. The same scenario will result in no system analysis at all for AAO, but will instead result in a large number of nonlinear inequality constraints in the optimization. This same scenario results in an intermediate state for IDF - an equilibrium equality constraint and a single analysis evaluation are required for each behaviour variable that promotes coupling.

A large factor that is affected by this topic of "problem dynamic" is the use of move limits within the optimizer. CASCADE has been known to generate treacherous design spaces that are cluttered with a multitude of local minima, any of which may contain the initial design point. Often times, very large move limits are required to retreat such a scenario. Because MDF (and to a lesser degree, IDF) has a distinct analysis step, large move limits must be handled with caution. If a design manager allows the design variables to change by a large degree on a given optimization iteration, the ensuing system analysis may see large changes in the behaviour variable magnitudes. This may cause the initial design on the subsequent optimization cycle to be infeasible, from which the optimizer might not be able to recover, depending on the degree of infeasibility. The authors' trial executions of the test systems during both "preliminary" and "verification" testing solidified these statements.

8.2 Results from "preliminary" testing (Section 7.1)

A major area of discussion involves the "goodness" of the results attained. The primary figure of merit for the test results is the final value of the objective function. Figures 8 and 10 plot the objective function histories of the first and second test systems, respectively. In both of these test systems, the MDF strategy, which has the fewest optimization variables, attains the lowest final objective function, and reaches its final solution the most quickly. The AAO strategy, which has the largest number of optimization variables, tends to lag towards its final solution much more slowly. Not surprisingly, the IDF strategy tends to follow a path which is intermediate to the extremes of the MDF and AAO strategies.

In fact, the MDF strategy attains the best final objective function value for test systems 3, 4, and 5 as well; refer back to Table 2. In all but the fifth test system (where AAO slightly outperforms IDF), the IDF strategy attains the next best final solution, followed closely by the final solution of the AAO strategy. Seemingly, for the types of problems generated by CASCADE (highly nonlinear and nonconvex), the solution strategy with the smaller number of optimization variables tends to attain the greatest improvement in the objective function.

Total execution times for the five test systems tended to follow the same trend - MDF attained its solution the fastest, followed by IDF and AAO. (Refer to Fig. 11). For the MDF solution strategy (FULL iterative system analysis), total execution times for the five test problems were approximately 0.5, 1.1, 22.9, 57.9, and 98.2 seconds, respectively, where the corresponding number of optimization variables for each test problem is 6, 11, 40, 45, and 40, respectively. (Refer to Table 2). For the AAO solution strategy (NO system analysis), total execution times were approximately 2.0, 9.1, 64.2, 261.0, and 3098.9 seconds for the five test systems, corresponding to 12, 20, 60, 90, and 140 optimization variables, respectively. One must keep in mind that these execution times are clock times required for multidisciplinary design *simulations*. To attain an understanding of the total time required to attain a solution for a real-life system (of the same coupling structure that is being simulated by CASCADE), one would have to factor in the relative times required for analysis and objective function calculations for the real-life system itself. Analysis calculations are likely to be the largest factor in terms of total execution time, by a considerable margin, which is not being reflected in the CASCADE test problem simulations. These issues were initially addressed in Section 7.3, and will be further discussed in Section 7.4.

8.3
Results from "verification" testing (Section 7.2)

Having performed both the preliminary testing (Section 7.1) and the subsequent verification testing (Section 7.2), and upon observing the tabular (Tables 3–6) and graphical (Figs. 12–16) results corresponding to the verification testing, numerous statements can be made with a greater deal of confidence. As system size increases, the likelihood that either IDF or AAO will outperform MDF (in terms of final objective function attained), decreases. For the first system size, where the optimization variable count is low for all solution strategies (MDF = 6; IDF = 11; AAO = 12), all solution strategies perform fairly well. MDF attains the best final solution for five of the nine system instances, and IDF and AAO each attain the best final solution for two of the nine instances. System size 2 is slightly larger, as is the corresponding optimization variable count: (MDF = 9; IDF = 17; AAO = 18). Here, MDF attains the best final solution for seven of the nine system instances, and IDF and AAO each attain the best final solution for one of the nine instances. Aside from a single system instance in system size 3, MDF attains the best final solution in all remaining instances of system sizes 3, 4, and 5, through which the optimization variable count continues to increase in size.

Some trends regarding the correlation of the solution strategies with system volatility factor are worthy of note. For system size 1, the smallest system size tested, the AAO strategy attains the lowest final solution in two of the nine system instances; both of those instances are low volatility (SVF = 1) system instances. The IDF strategy also attains the lowest final solution in two of the nine system instances; both of those instances are high volatility (SVF = 5) system instances. The MDF strategy attains the lowest final solution in all three of the intermediate volatility (SVF = 3) system instances, and one each of the low and high volatility system instances. As system size increases, these trends remain somewhat evident, but quickly diminish. For system size 2, the AAO strategy attains the lowest final solution in one of the nine system instances; which is again a low volatility (SVF = 1) system instance. For system size 3, the IDF strategy attains the lowest final solution in one of the nine system instances; this instance is again a high volatility (SVF = 5) system instance. As system size increases to system sizes 4 and 5, the MDF strategy attains the lowest final solution, regardless of system volatility.

Not only does the likelihood that MDF will outperform the IDF and AAO strategies grow with the size of the system, but so too does the amount by which MDF outperforms them. Table 5 appears to be a clear indicator of this fact. When comparing these results "down the rows", it is seen that average normalized improvement (which was explained in the final paragraph of Section 7.2) for MDF is calculated as 0.97 for the first system size, grows to 0.99 for the third system size, and maximizes to 1.0 for the fourth and fifth system sizes. For the IDF solution strategy, the average normalized improvement is calculated as 0.86 for the first system size, decreases to 0.67 for the third system size, and steadily decreases to 0.21 for the fifth system size. For the AAO solution strategy, the average normalized improvement is calculated as approximately 0.61 for the first three system sizes, decreases to 0.27 for the fourth system size, and drops to 0.22 for the fifth system size. If these results are instead compared "across the columns", and if the average normalized improvement is to be perceived as a measure of reliability for attaining the greatest improvement in objective function, one sees the following. For all but the fifth system size (where AAO slightly outperforms IDF), MDF is the most reliable strategy, followed by IDF, followed by AAO.

Table 6, though based on smaller subsections of data, reveal some noteworthy trends as well. The "SVF = 1" data shows that all three solution strategies perform comparably at low system volatility, and at smaller system sizes (sizes 1 and 2). As system size increases, MDF begins to outperform the other strategies by a steadily increasing amount. The general tendency is that IDF outperforms AAO, but this is not always the case. The "SVF = 3" data, which presents intermediate system volatility data, supports this possibility. Here however, a greater disparity between the performance of the solution strategies is seen at the smaller system sizes. Finally, the "SVF = 5" data demonstrates that at high system volatility, the performance of the solution strategies becomes more unpredictable. MDF continues to perform well for all system sizes, and is nearly matched by the performance of IDF at the smallest system size. IDF and AAO perform "respectably" against MDF for the second and third system sizes, but are greatly outperformed by MDF for the larger system sizes (4 and 5).The next section will focus on the downside of implementing a particular solution strategy; a downside that must be weighed against the existing potential for design improvement. The downside, of course, is the cost incurred while implementing a particular solution strategy.

8.4
Results from "cost-based" testing (Section 7.3)

Clearly, for the types of systems generated by CASCADE, the MDF solution strategy has shown itself to be the most reliable, regardless of system size or system volatility, for attaining the greatest design improvement. However, the results of Section 7.3 demonstrate that the cost incurred in attaining this improvement is tremendous. Analysis cost is typically ten times (or greater) larger for MDF than for IDF. In general, the factor by which analysis cost for MDF exceeds that for IDF is a function of how many iterations are required to converge the system of coupled nonlinear analysis equations each design cycle. To a lesser degree, this factor is also a function of what percentage of the system behaviour variables are not coupling variables. When it comes to analysis cost, AAO is undoubtedly the most attractive solution strategy – there is absolutely no *explicit* analysis cost. As the average cost of an analysis evaluation increases between the cost scenarios, the total analysis cost increases for MDF and IDF, but does so in proportion.

Objective function cost tends to be comparable for all three solution strategies, and small when compared to the other costs incurred during solution strategy implementation. Being that AAO has the largest associated optimization problem, it tends to incur the greatest objective function cost. This is not always the case; for test system 2, AAO actually incurs the smallest objective function cost. The objective function cost is clearly a function of the dynamic of the problem being solved - the amount of design improvement that is possible by the particular solution strategy being implemented. This is clearly a function of the initial starting design point. As the average cost of an objective function evaluation increases between the cost scenarios, the total objective function cost increases for all three solution strategies, but does so in proportion.

Constraint function cost exhibits the most interesting and noteworthy behaviour when comparing the solution strategies and cost scenarios. For MDF, the constraint cost is solely a function of inequality constraint evaluation cost, which remains constant for all three cost scenarios. For IDF, the constraint cost is a function of both inequality constraint evaluation cost (constant) and equilibrium constraint cost for each coupling variable. Recall that the latter cost has been approximated to be equal to 10% of the analysis cost for each analysis equation (behaviour variable) that corresponds to each coupling variable – this quantity changes between cost scenarios. Similarly, for AAO, the constraint cost is a function of both inequality constraint evaluation cost (constant) and the cost of each analysis equation posed as an equality constraint. Again, this latter quantity changes between cost scenarios. Hence, constraint cost is the only of the three cost quantities that changes disproportionately between the cost scenarios. For example, for test system 1, cost scenario 1, AAO has the largest constraint cost, followed by that of MDF, which is about half as large, followed by IDF, which is about one-third as large as that of AAO. For cost scenario 2, where inequality constraint cost remains constant but analysis cost (on average) increases five-fold (refer to Table 8), the AAO constraint cost becomes almost three times larger (than its cost scenario 1 value), MDF constraint cost remains constant, and IDF constraint cost nearly matches that of MDF. Finally, for cost scenario 3, the AAO constraint cost doubles from that of its cost scenario 2 value, MDF constraint cost again remains constant, and IDF constraint cost overtakes that of MDF by approximately 25%. For this cost scenario, the AAO constraint cost is almost ten times larger than that of IDF, which has the second highest total constraint cost.

Total cost is a function of all three costs that have been discussed thus far. Total cost was found to be highest for all test systems and all cost scenarios for the MDF strategy, and usually by a substantial margin. This is due almost entirely to the enormous iterative analysis cost that is incurred every design cycle. Analysis cost is also found in the IDF (as the analysis cost) and AAO (as a vast portion of the constraint cost) strategies, but recall that analysis cost is noniterative in nature in IDF and AAO, resulting in a lower summated final cost. Objective function cost was found to have a miniscule effect on the total cost for all test problems and cost scenarios. Constraint cost was the only cost quantity that increased disproportionately between the cost scenarios. As explained previously, this is because the constraint cost for the IDF and AAO strategies is a function of both the constant inequality constraint evaluation cost, and the nonconstant analysis evaluation cost, which appears by virtue of the equilibrium constraints and the analysis-based equality constraints, respectively.

For the first and third test systems, total cost was higher for AAO than it was for IDF. This came as a mild surprise to the authors, but on closer observation, began to make sense. For the first test system, MDF and AAO performed comparably in terms of objective function improvement. IDF showed improvement, but substantially less than that of the other two strategies. Hence, IDF likely converged upon its inferior solution relatively quickly, with a great deal fewer evaluations (both analysis and optimization) expended. For the third test system, IDF slightly outperformed AAO in terms of objective function improvement. Here again, the authors speculate that IDF reached its final solution (which looks to be a stingy local optimum) rather quickly, and with fewer total evaluations, than did AAO, which achieved roughly the same final solution, but which has more optimization variables and a more complicated design space than does IDF. In both cases outlined above, IDF would hence have a lower associated total cost than would AAO.

A final observation to be made involves objective function cost, which appears to be a good gauge for determining which solution strategy will incur the lowest total cost. For the first and third test systems, the objective function cost (and hence the total number of objective function evaluations) for the AAO strategy is almost twice as large as that for the IDF strategy. In both test systems, the total cost for the AAO strategy approximately doubled that of the IDF strategy. For the second test system, the objective function cost for the IDF strategy was slightly larger than that for the AAO strategy. For this test system, the total cost of the IDF strategy was slightly larger than that for the AAO strategy. As previously explained, MDF was found to have the largest total cost, often by a substantial margin, regardless of the objective function cost.

8.5
Closing observations

It is the authors' theory that for any given MDO problem at some initial starting point, an improved local (or hopefully global) optimum solution can be found, regardless of the solution strategy that is applied. The real issue is the "ease" with which the best found solution is attained, with each of the strategies. To this end, MDF consistently arrived at the "best found" design point on one of the first few trial executions, and with minor modifications of the default optimizer settings, or no modifications at all. The IDF and AAO solution strategies were quite the contrary - many more trial executions were required to attain the corresponding "best found" solution for each of these strategies. Moreover, substantial modifications to the default optimizer settings were required. Some of the default optimizer parameters which were commonly adjusted (during the "preliminary" testing phase only) include: relative move limits, relative and absolute finite difference step sizes, the value of the penalty parameter for equality constraints (IDF and AAO only), maximum iterations and convergence criteria at both the optimizer and strategy levels, and the binary switch for variable scaling. Variable scaling worked consistently well for the MDF solution strategy, whose optimization variables consist solely of design variables. Often times, the IDF and AAO strategies required that variable scaling not be used, especially for larger system sizes. This is likely because these strategies have both design and behaviour variables acting as optimization variables and numerous additional equality constraints resulting from the analysis equations, which renders the scaling of the design problem an impossible task.

9
Conclusions

This research effort has presented a numerical comparison of the MDF, IDF, and AAO solution strategies across simulated multidisciplinary design systems of varying size and complexity. The multidisciplinary test systems have been generated by CASCADE, a robust simulation tool. From the results that have been attained, some general conclusions can be drawn. Given the ability to pose a multidisciplinary design problem three different ways, a design manager has three distinct means for solving the design problem, which should only be to his or her advantage. Each of the three solution strategies exemplifies a unique problem dynamic, greatly dependent on the initial design point, and in most cases, on the settings of the optimizer. Numerous figures of merit must be considered when choosing an appropriate solution strategy; the primary of which is design improvement. The MDF strategy has been found to arrive at the "best found" solution for a vast majority of the test problems in this effort. AAO usually attains final solutions that are vastly improved (over the initial design) without any form of costly iterative system analysis, but at the expense of a complex design space that is cluttered with nonlinear equality constraints. IDF has shown itself to be a good trade-off between the extremes of MDF and AAO; fewer analysis evaluations than MDF, and (equilibrium) equality constraints that are simpler, and usually fewer in quantity than those of AAO, depending on the coupling nature of the system. The IDF was typically found to outperform the AAO strategy, in terms of objective function improvement.

The reader must realize that the results that have been presented represent MDO problem simulations, and must be interpreted as such. In a real-world MDO problem, total solution times for the MDF strategy would likely have been the longest by a considerable margin (due to the many costly iterative analysis evaluations that it requires), followed by IDF (fewer, noniterative analysis evaluations) and AAO (no iterative analysis evaluations).

The cost-based testing support this line of speculation. MDF was found to be the most costly solution strategy in all three cost-based test systems, usually by a factor of ten or more. The cost comparison between IDF and AAO is not straightforward. AAO was found to be twice as costly as IDF for two of the three cost-based systems tested, and outperformed IDF (in terms of objective function improvement) in one of these two systems. AAO was found to be slightly less costly than IDF in one cost-based test system, but was slightly outperformed by IDF in this test system. Finally, the authors reason that an improved local (or hopefully global) optimum solution can be found, regardless of the solution strategy used. The ultimate goal of the design manager is to utilize the solution strategy which will achieve the "best found" design with the least difficulty. To this end, the MDF strategy was found to constantly achieve its "best found" solution with minimal modification of optimizer settings, and with the smallest number of trial executions.

Acknowledgements The authors would like to acknowledge the partial funding support of The National Science Foundation from the following grants: NSF DMI9553210 and NSF DMI9622314.

References

Balling, R.J.; Sobieszczanski-Sobieski, J. 1994: Optimization of coupled systems: A critical overview of approaches. *AIAA Paper 94-4339*

Balling, R.J.; Wilkinson, C.A. 1997: Execution of multidisciplinary design optimization approaches on common test problems. *AIAA J.* **35**, 1–9

Bloebaum, C.L.; Hajela, P.; Sobieszczanski-Sobieski, J. 1990: Non-hierarchic system decomposition in structural optimization. *3-rd USAF/NASA Symp. on Recent Advances in Multidisciplinary Analysis and Optimization* (held in San Francisco, CA)

Cramer, E.J. *et al.* 1992: On alternative problem formulations for multidisciplinary design optimization. *4-th AIAA/NASA/ISSMO Symp. on Multidisciplinary Analysis and Optimization* (held in Cleveland, OH)

Cramer, E.J. *et al.* 1994: Problem formulation for multidisciplinary optimization. *SIAM J. Optim.* **4**, 754–776

Haftka, R.T. 1985: Simultaneous analysis and design. *AIAA J.* **23**, 1099–1103

Haftka, R.T.; Sobieszczanski-Sobieski, J.; Padula, S.L. 1992: On options for interdisciplinary analysis and design optimization. *Struct. Optim.* **4**, pp. 65-74

Hulme, K.F.; Bloebaum, C.L 1996: Development of CASCADE - A multidisciplinary design test simulator. *6-th AIAA/NASA/ISSMO Symp. on Multidisciplinary Analysis and Optimization* (held inBellevue, WA)

Hulme, K.F.; Bloebaum, C.L. 1997: Development of a multidisciplinary design optimization test simulator. *Struct. Optim.* **14**, 129–137

Hulme, K.F.; Bloebaum, C.L. 1998: A comparison of solution strategies for simulation-based multidisciplinary design optimization. *7-th AIAA/NASA/ISSMO Symp. on Multidisciplinary Analysis and Optimization* (held in St. Louis, MO)

Sobieszczanski-Sobieski, J. 1982: A linear decomposition method for optimization problems - Blueprint for development. *NASA Technical Memorandum 83248*

Sobieszczanski-Sobieski, J. 1990: The sensitivity of complex, internally coupled systems. *AIAA J.* **28**, 153–160

Vanderplaats, G. 1985: *ADS - A FORTRAN program for automated design synthesis. Version 1.10 (user's manual)*. Santa Barbara, CA: University of California