ICASE

OBSERVATIONS REGARDING USE OF ADVANCED CFD ANALYSIS, SENSITIVITY ANALYSIS, AND DESIGN CODES IN MDO

Perry A. Newman
Gene J.-W. Hou
Arthur C. Taylor, III

National Aeronautics and
Space Administration

Langley Research Center
Hampton, Virginia 23681-0001

OBSERVATIONS REGARDING USE OF ADVANCED CFD ANALYSIS, SENSITIVITY ANALYSIS, AND DESIGN CODES IN MDO

Perry A. Newman
NASA Langley Research Center
Hampton, VA 23681-0001

Gene J.-W. Hou[*][†] *and Arthur C. Taylor, III*[†]
Old Dominion University
Norfolk, VA 23529-0247

Abstract

Observations regarding the use of advanced computational fluid dynamics (CFD) analysis, sensitivity analysis (SA), and design codes in gradient-based multidisciplinary design optimization (MDO) reflect our perception of the interactions required of CFD and our experience in recent aerodynamic design optimization studies using CFD. Sample results from these latter studies are summarized for conventional optimization (analysis-SA codes) and simultaneous analysis and design optimization (design code) using both Euler and Navier-Stokes flow approximations. The SA codes is greater than that required for design codes. Thus, an MDO formulation that utilizes the more efficient design codes where possible is desired. However, in the aerovehicle MDO problem, the various disciplines that are involved have different design points in the flight envelope; therefore, CFD analysis-SA codes are required at th aerodynamic "off design" points. The suggested MDO formulation is a hybrid multilevel optimization procedure that consists of both multipoint CFD analysis-SA codes and multipoint CFD design codes that perform suboptimizations.

1 Introductory Remarks

The focus of this work is on only those techniques that are applicable to advanced (high-fidelity) computational fluid dynamics (CFD) and that are extendable to multidisciplinary design optimization (MDO) for realistic three-dimensional (3–D) aerovehicles. Table 1 briefly summarizes this research and gives the chief advantages and disadvantages of each technique.

Table 1. *Summary of Techniques Used in Present Focus*

TECHNIQUE	ADVANTAGE	DISADVANTAGE
Advanced CFD	Required physical fidelity	Nonlinear and high costs
Multiblock (or unstructured)	Required geometric complexity	Higher costs
Multigrid acceleration	Efficient algorithm	Tedious code
Gradient-based optimization	gradient information in math models	Local vs. global minima
Automatic differentiation	Accurate and robust	Efficiency questions
Incremental iterative method	Consistent, efficient, and versatile SA algorithm	Not automatic

Application of these techniques to advanced CFD codes has been proposed and discussed in [1]–[5]. In [1] and [2], the incremental iterative method (IIM) for calculation of sensitivity derivatives (SD's) is discussed. In [3], the application of automatic differentiation (AD) to obtain SD's from a 3–D thin-layer Navier-Stokes code is demonstrated. References [4] and [5] are recent summaries of the studies of this work; the combination of AD and IIM to efficiently obtain consistent discrete SD's from a two–dimensional (2–D) thin-layer Navier-Stokes code is demonstrated in [6]. The AD tool ADIFOR (AD of Fortran) of [7] and [8] has been used throughout this work. Symbols and acronyms are defined as introduced in the text.

The equations summarized in this paragraph can be found in greater detail in [1]–[6]. The conservation laws of compressible fluid flow, R, and aerodynamic functions, F, of interest can be expressed as

$$(1) \qquad R(Q(b), X(b), b) = 0 \qquad (Nonlinear\ state\ equation)$$

and

$$(2) \qquad F = F(Q(b), X(b), b) \qquad (Aerodynamic\ output\ function)$$

where Q is the vector of state (field) variables, X is the vector of computational grid coordinates, and b is the vector of design variables. Direct differentiation of Eqs. (1) and (2) with respect to the design variables yields

$$(3) \qquad R' \equiv \frac{dR}{db} = \frac{\partial R}{\partial Q} Q' + \frac{\partial R}{\partial X} X' + \frac{\partial R}{\partial b} = 0$$

and

$$(4) \qquad F' \equiv \frac{dF}{db} = \frac{\partial F}{\partial Q} Q' + \frac{\partial F}{\partial X} X' + \frac{\partial F}{\partial b}$$

where $Q' \equiv \frac{dQ}{db}$ and $X' \equiv \frac{dX}{db}$ are the flow and grid SD's with respect to the design variables. The introduction of an adjoint variable (vector) A associated with F gives

$$(5) \qquad \left(\frac{\partial R}{\partial Q}\right)^T A + \left(\frac{\partial F}{\partial Q}\right)^T = 0 \qquad (Costate\ equation)$$

and

$$(6) \qquad F' = \left(\frac{\partial F}{\partial X} + A^T \frac{\partial R}{\partial X} \right) X' + \frac{\partial F}{\partial b} + A^T \frac{\partial R}{\partial b}$$

where T denotes transpose. The IIM solution forms for Eqs. (1), (3), and (5) are, respectively,

$$(7) \qquad -\frac{\widetilde{\partial R^n}}{\partial Q} \Delta Q = R^n; \qquad Q^{n+1} = Q^n + \Delta Q \qquad (n = 1, 2, 3, ...)$$

$$(8) \qquad -\frac{\widetilde{\partial R}}{\partial Q} \Delta Q' = R'^m; \qquad Q'^{(m+1)} = Q'^{(m)} + \Delta Q' \qquad (m = 1, 2, 3, ...)$$

$$(9) \qquad -\left(\frac{\widetilde{\partial R}}{\partial Q} \right)^T \Delta A = \left(\frac{\partial R}{\partial Q} \right)^T A^n + \left(\frac{\partial F}{\partial Q} \right)^T; \qquad A^{n+1} = A^n + \Delta A \qquad (n = 1, 2, 3, ...)$$

where n and m are iteration indices and $\frac{\widetilde{\partial R}}{\partial Q}$ denotes an "approximate operator of convenience." The first-order SD's (F') are obtained from either Eq. (4) with Q' and X' or Eq. (6) with A and X'. Second-order SD's are also discussed in [6] but not used in the present work.
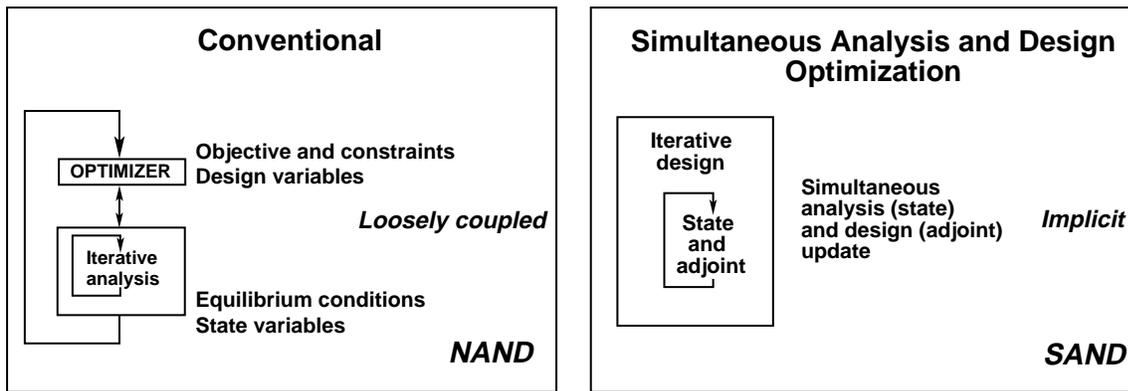
Calculation of SD's via finite differences (FD) for large sets of iteratively solved nonlinear equations can be expensive and inaccurate. A comparison of SD ratios for lift, drag, and pitching moment with respect to a geometric design variable for transonic turbulent flow about an airfoil [9] is shown in Table 2. The denominator in these SD ratios is the value of the SD via AD at relative residual, R/R_1, and residual derivatives, $\left(\frac{dR}{db} \right) / \left(\frac{dR}{db} \right)_1$, reductions of 10^{-6}. If the FD results (SD$_{\text{FD}}$) agreed with the AD results (SD$_{\text{AD}}$),

Table 2. *Sensitivity Derivative Ratio Comparisons: A Turbulent Viscous Example*

$\frac{R}{R_1}$	$\frac{SD_{FD}}{(SD_{AD})_{10^{-6}}}$	% change of design variable, b					$\frac{\left(\frac{dR}{db} \right)}{\left(\frac{dR}{db} \right)_1}$	$\frac{SD_{AD}}{(SD_{AD})_{10^{-6}}}$
		10	1	0.1	0.01	0.001		
10^{-3}	lift	0.937	0.926	0.040	—	—	10^{-3}	0.983
	drag	4.862	1.819	4.383	—	—		1.153
	moment	1.035	2.097	−0.099	—	—		0.989
10^{-4}	lift	1.043	0.973	0.873	−0.018	—	10^{-4}	0.999
	drag	3.792	2.585	7.883	29.154	—		1.007
	moment	1.139	0.944	1.129	0.385	—		0.999
10^{-5}	lift	—	0.985	1.039	0.973	−0.003	10^{-5}	0.999
	drag	—	1.567	−0.159	6.986	28.851		1.000
	moment	—	0.995	1.188	1.269	0.397		0.999
10^{-6}	lift	—	1.003	0.992	1.112	1.331	10^{-6}	1.000
	drag	—	1.507	0.991	−0.904	2.804		1.000
	moment	—	1.006	1.003	1.270	1.680		1.000

all table entries would be like those in the last column (i.e., essentially unity). The table shows that at a given convergence level for R, different output functions may require different design variable step sizes (and vice versa) for the SD$_{\text{FD}}$. The similarly shaded areas show comparable SD accuracies for the FD and AD approaches.

Recent overviews of optimization approaches for coupled systems (disciplines) are given, for example, in [10] and [11]; the present single-discipline optimization observations will be discussed in terms similar to those introduced in [10]. Conventional optimization utilizes analysis (and perhaps also sensitivity analysis (SA)) codes in conjunction with an optimization code, as shown in Fig. 1(a); when an analysis is an iterative one, it must be run to reasonably well-converged solutions many times. In [10], this method is called the nested analysis and design (NAND) approach; the optimizer and analysis codes are loosely coupled. At the other extreme is the simultaneous analysis and design (SAND) approach in which the design and state variables are updated together. For an iterative (state) analysis code, then, the design variable updates are made within the iterative analysis loops; the optimizer and analysis are implicitly coupled to produce an iterative design code as shown in Fig. 1(b). In reality, an entire range of design procedures exists between the two extremes of Figs. 1(a) and 1(b); these procedures differ only in the frequency at which the iterative analysis and optimization interact.

Conventional	Simultaneous Analysis and Design Optimization
OPTIMIZER — Objective and constraints, Design variables	Iterative design
Loosely coupled	Simultaneous analysis (state) and design (adjoint) update — Implicit
Iterative analysis — Equilibrium conditions, State variables	State and adjoint
NAND	SAND

(a) Nested analysis and design (NAND).   *(b) Simultaneous analysis and design (SAND).*

Fig. 1. *Approaches for an optimization with iterative (analysis) state equation.*

In the next few sections, this paper addresses recent aerodynamic single-point design optimization studies using both NAND and SAND approaches, with a comparison of computational requirements; proposed aerodynamic multipoint aerodynamic design optimization approaches; and, finally, a suggested MDO approach.

2 CFD Single-Point Design Optimization Results

The results discussed in this section are for recent single-discipline (CFD) single-point design optimization studies; details have been reported elsewhere as noted in the quoted references. Both NAND and SAND approach results have been obtained, although generally not for identical problems. Even though formal optimization procedures have been used, these studies produce design improvements as opposed to optimum solutions, probably because of the accepted convergence levels in the required iterative solutions for the nonlinear CFD equations.

2.1 Conventional Optimization: NAND with SA

The conventional optimization study results discussed here are for aerodynamic shape optimization using the NAND approach with SA based on analytical SD's. The High-Speed Civil Transport (HSCT) 24E design improvement studies are based on a 3–D marching (supersonic) Euler CFD code, and results are reported in [12]. The transonic turbulent airfoil study results are based on a 2–D thin-layer Navier-Stokes CFD code, and results are reported in [9].

2.1.1 HSCT Design Improvement. Initial results for aerodynamic shape optimization studies that investigate the feasibility of using a 3–D supersonic Euler code with an efficient SA capability are given in

[12] for a Mach 2.4 HSCT wing-body configuration. A comparison of typical nongeometric SD's of the force and moment coefficients with respect to Mach number, angle of attack, and yaw angle is given in [13]. The IIM and FD SD's agree to four significant digits; the IIM results are computationally less expensive to obtain. Typical aerodynamic SD's of the force and moment coefficients with respect to wing geometry parameters obtained with the IIM are also accurate and computationally less expensive than those obtained by efficient FD. A comparison of these geometric SD's with respect to wing-section thickness, twist, and camber, as well as wing planform and flap deflections is given in [12]. Again, the agreement between the IIM and FD results is very good, and the IIM results are computationally less expensive to obtain.

The computational flowchart for the aerodynamic shape optimization studies is shown in Fig. 2, where the (outer) shape design iteration loop is shown. In these studies, extensive use was made of solution restart files; these inner loops are not shown in Fig. 2. The shape design loop starts at the upper left with the automated surface-shaping and volume-grid generation codes, which are discussed in [14] and [15]. These codes are differentiated with ADIFOR ([7] and [8]) to provide the grid SD's (X') with respect to approximately 100 (wing) geometric design variables, as discussed in the appendix of [12]. Both the grid (X) and its SD's (X') are required because the geometric (shape) design variables determine the vehicle surface and its body-fitted computational grid. The marching Euler code is differentiated by hand $\left(\frac{\partial R}{\partial Q}, \frac{\partial R}{\partial X}, \text{and } \frac{\partial R}{\partial b}\right.$, and likewise for output functions $F\big)$ to construct the flow derivative code. The Automated Design Synthesis (ADS) program [16] is used for the present constrained optimization results; the sequential quadratic programming strategy, the modified method of feasible directions optimizer, and the Golden Section line search options have been selected. Evaluation of both function and first-order derivatives (SD's) is given to the ADS code. Because the SD via the IIM are essentially analytical derivatives, this combination of methods in ADS gives the most consistent optimization results. However, many function evaluations are required by the selected search procedure. Increments in the shape design variables, denoted "Del geom input" in Fig. 2, are returned to the surface shaping code to start the next design iteration.
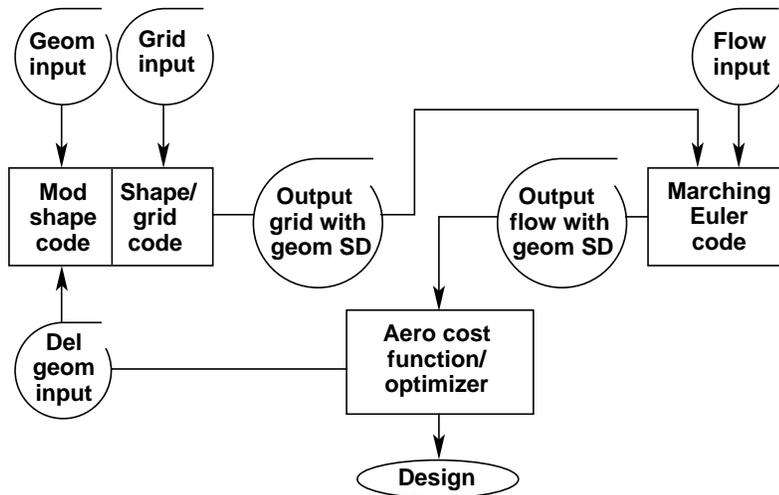

Fig. 2. *Flowchart for aerodynamic shape optimization feasibility studies.*

Sample results from two of the eight optimization studies reported in [12] are discussed here. The HSCT 24E filleted wing-body configuration generated by NASA Langley Research Center is the baseline shape; the flow conditions are Mach number $M_\infty = 2.4$, angle of attack = 1°, and yaw angle $\beta = 0°$. Convergence of both the nonlinear iterative flow analysis and the linear iterative SA is to a relative residual reduction of 6 orders of magnitude for all required solutions.

For the wing-section thickness design improvement study, initial and final thickness distributions are shown in Fig. 3. The 15 design variables consist of 5 parameters each at the wing root, break, and tip locations. The wing is linearly lofted from root to break and from break to tip to supply thickness information at all other wing stations. The objective function is drag minimization, with the wing root bending moment and lift constrained to their baseline values; that is, minimize $\frac{C_x}{C_{x_0}}$ subject to $\frac{C_{M_x}}{C_{M_{x_0}}} \leq 1.0$

and $\frac{C_Z}{C_{Z_0}} \geq 1.0$. The baseline drag is decreased by about 10 percent, and both constraints are active. This improvement is obtained in 8 optimization steps, which requires 117 function evaluations and 8 gradient evaluations; the Cray-2 run time is about 1.2 hours. For 6 of the 15 design variables, the side constraints are active (within 5 percent of the specified bounds, which were arbitrarily taken for the thickness variables to be ±50 percent of the baseline values). For supersonic flow considerations alone, the wing would be expected to become thinner, as shown in Fig. 3.
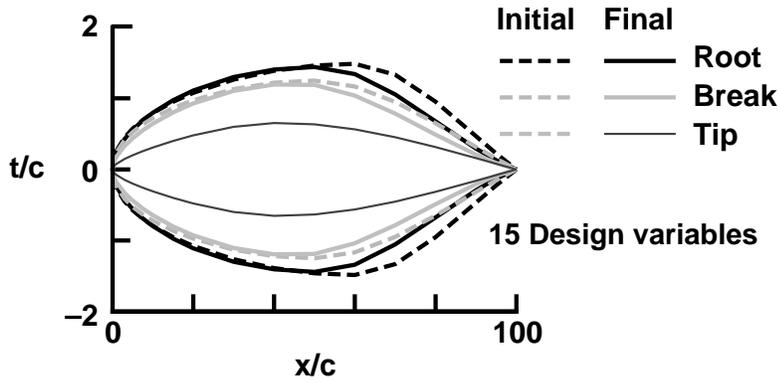

Fig. 3. *Wing section thickness distributions for HSCT 24E design improvement.*

For the wing planform design improvement study, initial and final planforms are shown in Fig. 4. The five design variables are the wing root, break, and tip chord lengths and the streamwise locations of both break and tip section leading edges. The objective is to minimize $-\frac{C_Z}{C_{Z_0}}$ subject to $\frac{C_{M_X}}{C_{M_{X_0}}} \leq 1.0$ and $\frac{C_X}{C_{X_0}} \leq 1.0$. The baseline lift is increased by 5.5 percent, and the drag constraint is violated by 3.8 percent. Neither the wing bending-moment constraint nor any of the design-variable side constraints are active or violated. For supersonic flow considerations alone, the wing tip should be swept more than in the baseline HSCT 24E; Fig. 4 shows that the optimization procedure produces that result. At a Mach number of 2.4,
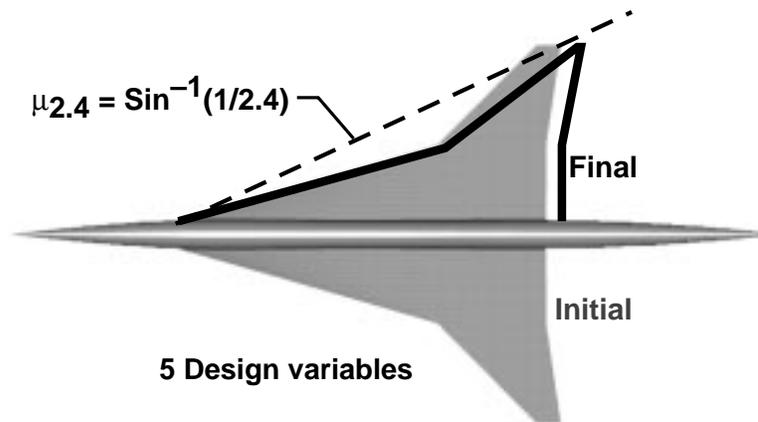

Fig. 4. *Wing planform shape for HSCT 24E design improvement.*

the Mach angle (μ) is 24.6°. The angle subtended by the wing-tip leading edge from the root leading edge is 25.9° for the baseline HSCT 24E and 23.8° for the final optimized planform. That is, the planform optimized for only supersonic flow lies behind the Mach cone.

2.1.2 Turbulent Transonic Airfoil Improvement. Initial results for aerodynamic shape optimization studies that investigate the feasibility of using a 2–D Navier-Stokes code with an efficient SA capability are given in [9] for a turbulent transonic airfoil. Comparisons of typical SD's of force and moment coefficients with respect to both flow and geometric variables are given in [6] for several methods of

5

calculation. The agreement between the IIM via AD and FD is good; the computational costs for the IIM using AD are comparable to or less than those for the FD.

The flowchart for this airfoil design improvement process is essentially that shown in Fig. 2; a more detailed flowchart with the restart loops indicated is given in [9]. The automated geometry and grid code TBGG (two-boundary grid generation) [17] is used for algebraically generated C grids around airfoils. The airfoil upper and lower surfaces are each represented by a linear combination of four orthonormalized polynomials [18]; their amplitude (or weights) are the shape design variables. The entire TBGG code is differentiated with ADIFOR ([7] and [8]) to obtain the grid SD's (X') with respect to these eight design variables. The CFD code ANSERS (algorithm for the Navier-Stokes equations based on a Riemann solver) [19] is a 2–D Navier-Stokes code based on an upwind cell-centered finite-volume formulation and simulates turbulence with the Baldwin-Lomax [20] algebraic turbulence model. This code is also differentiated with ADIFOR; however, the differentiation is done in parts so that an efficient IIM can be constructed for the SD's [6]. As in the previous example, the ADS program [16] is used for the present constrained optimization results. The sequential quadratic programming strategy, the modified method of feasible directions optimizer, and the Golden Section line search options are selected; evaluation of both the function and first-order derivatives is provided to the ADS code.

The design optimization problem summarized here (and fully discussed in [9]) is the maximization of the lift-to-drag ratio of an airfoil in a turbulent transonic flow. The incident flow is at a Mach number of 0.8, a chord Reynolds number of 5 million, and an angle of attack of 1°. Solutions are obtained on a C-type mesh of 257 (circumferential direction) × 67 (normal direction) points. In order to maintain the convexity of the airfoil around the leading edge, 10 constraints of positive curvature are imposed at equally spaced grid points on both surfaces for the first 5 percent chord.

The initial solution (Q) for the NACA 0012 airfoil is converged to a relative analysis residual reduction of 10 orders of magnitude, whereas for the SD (Q') the relative derivative residuals are converged 6 orders. These solutions were used as initial values for design optimization iterations. An optimal solution was obtained after three resubmittals to ADS; these resubmissions ensure that the optimization iterations have not terminated prematurely and are denoted by the black symbols on the objective function history plot of Fig. 5. The objective function C_ℓ/C_d has been increased to approximately 10 times its initial value in 58 design iterations. This optimization requires the same amount of central processing unit (CPU) time as required for approximately 36 well-converged single aerodynamic analyses. The strong shocks present in the initial design have been eliminated from the final design, as can be seen from the surface pressure distribution plots (C_p) in Fig. 5 and the flow-field Mach-number contour plots shown in Fig. 6. The airfoil has been thinned and cambered considerably, as would be expected in order to maximize the lift-to-drag ratio at transonic flow conditions.
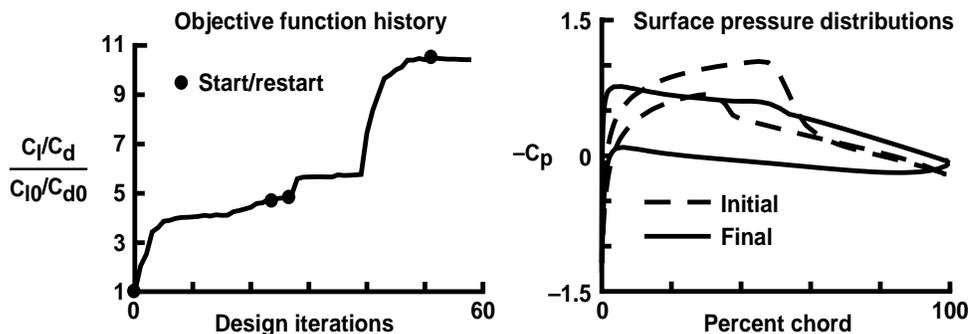

Fig. 5. *Objective function and surface pressure distributions for turbulent transonic airfoil improvement.*

In the results above, the relative residual reductions for both Q and Q' were set to 4 orders in the design optimization; additional solution iterations of Eq. (1) were required to improve the solution quality of the final design. However, a limited study of several cases to assess the effects of both flow analysis and SA convergence tolerances on the performance of the aerodynamic design optimization is given in [9]. Different combinations of flow and SA tolerances (differing by factors $10^{\pm1}$) result in different achieved objective functions (differing by 50 percent to 75 percent); these different combinations require different

total CPU times and produce different airfoils. Such different results are not unexpected in this gradient-based optimization procedure, because the (partially converged) values of aerodynamic functions and derivatives govern the (approximate) objective function surface and the (approximate) search direction on that surface. The different airfoils that result may also correspond to different local optima on the true (well-converged) objective function surface.
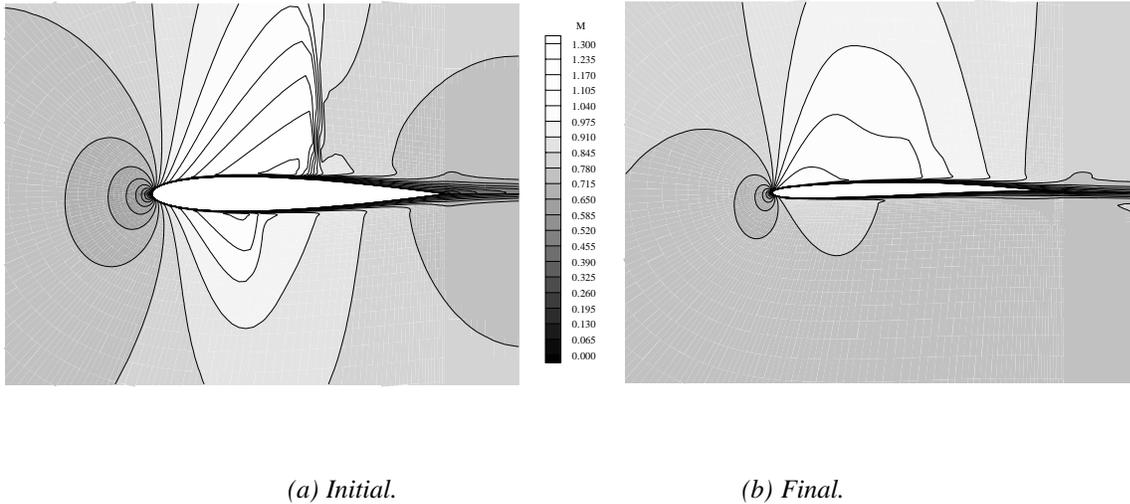

(a) Initial.   *(b) Final.*

Fig. 6. *Flow-field Mach number contour plots for turbulent transonic airfoil improvement.*

2.2 Simultaneous Analysis and Optimization: SAND

Several simultaneous analysis/optimization demonstration study results are discussed here for aerodynamic shape optimization using a SAND approach with an SA, based on analytical SD's or adjoints. The present procedure, using CFD, has been called SAADO (simultaneous aerodynamic analysis and design optimization). This procedure incorporates design optimization within CFD analysis to achieve a converged flow solution and an optimal design at the same time. For advanced 3–D CFD codes, an iterative solution of the linearized approximations to the nonlinear flow equations is required because of the large matrices involved. Design optimization is also iterative; the SAADO procedure interacts these two iterations (before either is converged), allowing a simultaneous relaxation of both the flow-field solution and the design optimization. Overall computational efficiency is achieved because expensive iterative solutions for nonoptimal design parameters are not converged (i.e., obtained). Initial demonstration studies are for quasi-one-dimensional (1–D) nozzle flows described by an Euler equation; the results are reported in [21] and [22]. The transonic turbulent airfoil results are based on a 2–D thin-layer Navier-Stokes CFD approximation and the results are reported in [23].

2.2.1 Quasi–1–D Nozzle Demonstration. The basic formulation and several variations of SAADO have been derived and successfully implemented for design optimization of quasi-1–D nozzles [21, 22]. The feasibility of SAADO is demonstrated for both supersonic and transonic flow described by a 1–D Euler equation. Optimization results for a supersonic nozzle design are shown in Fig. 7(a). Standard NAND optimization results with ADS software [16], for both one-sided FD and quasi-analytical (QA) [2] differentiation for SD's are compared with the SAADO results. The SAADO procedure produces a better objective minimization (which, in this case, should tend to zero) in 20 design cycles with fewer equivalent Newton-Raphson (NR) iterations and in less computer time than either of the standard optimizations. For the transonic nozzle, the residual (R) history plots in Fig. 7(b). show that SAADO requires approximately the same number of iterations to reach the optimal solution as is required for a single flow analysis. The oscillating nature of the SAADO residual history indicates that the incorporation of design changes alters the normal convergence pattern for flow analysis; these oscillations cease when the design changes cease.

In both nozzle designs, the objective is to match a prescribed velocity distribution throughout the length of the nozzle. The results are shown in Fig. 8.

Optimizer	ADS	ADS	SAADO
SD calculation	FD	QA	
Objective	8.8E-4	8.9E-4	8.9E-5
Equiv. NR iter.	174	102	80
Cray-2 (sec)	2.19	1.29	0.69
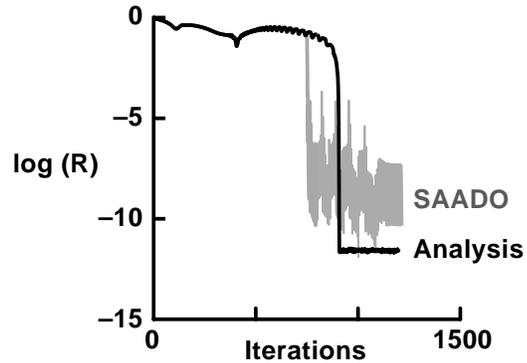
(a) Supersonic.

(b) Transonic.

Fig. 7. *Convergence results for SAADO quasi-1–D nozzle flow feasibility demonstration.*
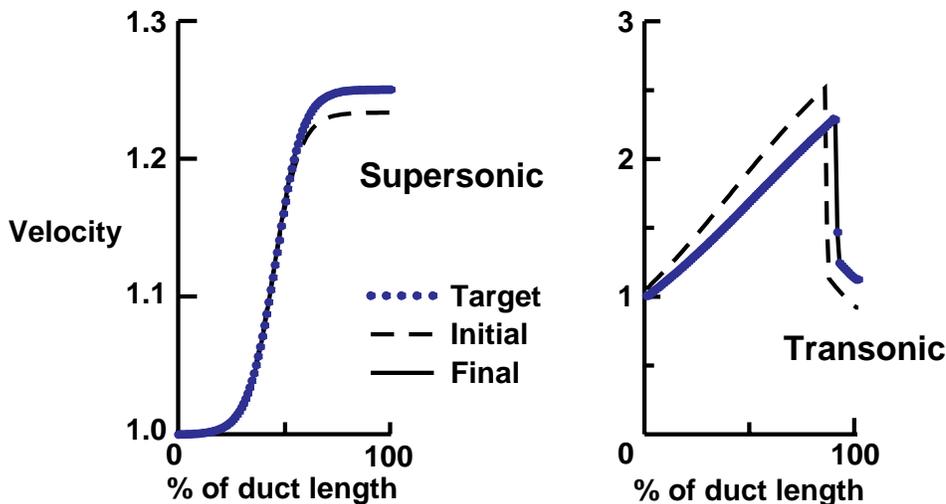
Fig. 8. *Nozzle velocity distributions for supersonic and transonic flow SAADO feasibility demonstration.*

The primary objective of the SAADO formulation is a significant reduction in the CPU time; the CPU time required for the repeated analyses in standard NAND optimization is impractical. The potential for SAADO is clearly demonstrated in these sample 1–D nozzle design optimization problems. For the supersonic flow case, the better solution is obtained by SAADO in 53 percent of the CPU time required by the NAND (QA for SD) approach; this solution time corresponds to the time required for approximately 9 equivalent analyses. For the transonic flow case, the SAADO approach obtains the design in the time required for 1 or 2 analyses.

2.2.2 Turbulent Transonic Airfoil Demonstration. The initial results for aerodynamic shape optimization with a 2–D SAADO procedure include both Euler and thin-layer Navier-Stokes CFD approximations. Results for the latter are reported in [23] and are briefly discussed here. The physical problem is described in section 2.1.2; however, airfoil thickness constraints have been added to account for a structural wing box. This 2–D SAADO procedure is constructed from the same grid-generation (TBGG [17]) and flow (ANSERS [19]) codes briefly discussed in section 2.1.2. Both hand differentiation (Euler, adjoint)

and ADIFOR (Navier-Stokes, direct differentiation) have been used to generate the required sensitivity equations for the SAADO versions that have been tested.

Presently, the 2–D form of the SAADO procedure for a thin-layer Navier-Stokes code considers both airfoil-shape and flow-field variables as independent (design) variables and treats the flow equations as equality constraints. This large number of variables and constraints is reduced by solving a set of inexact sensitivity equations; these equations and the flow analysis are solved together in IIM form. The design optimization problem is to maximize the lift-to-drag ratio C_ℓ/C_d of an airfoil in transonic turbulent flow, subject to 25 geometric constraints: 20 on the surface curvature imposed from the leading edge to 5 percent chord and 5 on the airfoil thickness imposed from 20 to 60 percent chord. Airfoil upper and lower profiles are each represented by a linear combination of four orthonormalized polynomials, as in [18]. The initial values of these eight polynomial weighting coefficients (the shape design variables) are those for the NACA 0012 airfoil.

Sample results for the SAADO approach to improving the lift-to-drag ratio at a Mach number of 0.8, an angle of attack of 1°, and a chord Reynolds number of 5 million, starting from the NACA 0012 airfoil, are obtained on a C-type mesh of 127×33 points. Figure 9 shows the objective function (versus
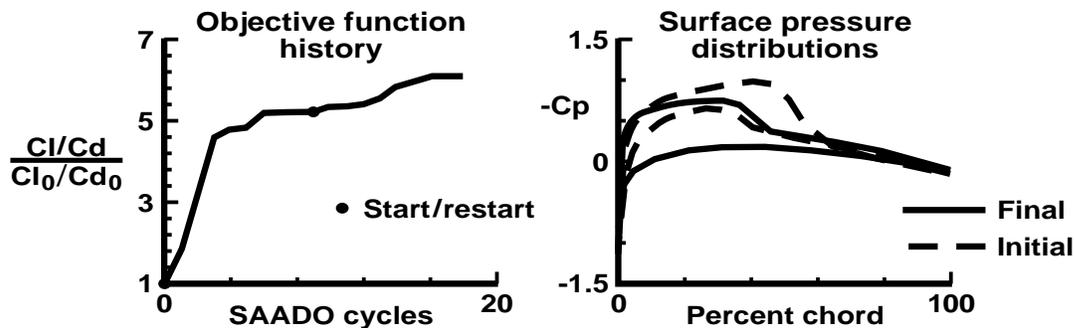

Fig. 9. *Objective function and surface pressure distributions for turbulent transonic airfoil SAADO demonstration.*

number of SAADO cycles) history and surface pressure (coefficient, $-Cp$) distributions on the initial and final airfoil surfaces. In 18 SAADO cycles (design variable updates), the flow analysis convergence level was reduced from 10^{-2} to 10^{-6}, which produces an airfoil with approximately 500 percent improvement in the objective function. Two thickness constraints, (at 20 and 60 percent chord) were active. These results show that the shock wave on the lower surface of the airfoil has been eliminated; the shock wave on the upper surface has been weakened considerably. These different shock wave patterns can also be seen in the initial and final flow-field Mach number contour plots, which are shown in Fig. 10.

The computational time required for this SAADO case is approximately the same as 22 flow analyses converged to a relative residual error of 10^{-6} on this 127×33 grid. For the inviscid (Euler) approximation, the relative SAADO time has been reduced further. The set of eight inexact aerodynamic sensitivity equations is replaced by three previously obtained, hand-differentiated adjoint equations reducing the computational time by approximately one-third. For a single adjoint equation, which corresponds to the single output function C_ℓ/C_d, the estimated reduction in computational time would be by a factor of approximately 3. This reduction is not currently feasible for the Navier-Stokes code because the ADIFOR tool does not provide a computationally efficient adjoint equation code. Nevertheless, the feasibility for implementation and use of the SAADO procedure on a 2–D thin-layer Navier-Stokes code in transonic turbulent airfoil design improvement has been demonstrated; ADIFOR was used to generate the derivative code.

2.3 Comments on NAND versus SAND Optimization.

A number of previous formulations that differ from SAADO have also involved efficient SAND methodologies for aerodynamic design optimization. The works of Rizk [24], Campbell [25], Drela [26], and Young et al. [27], for example, either incorporated simple design modification rules in the aerodynamic
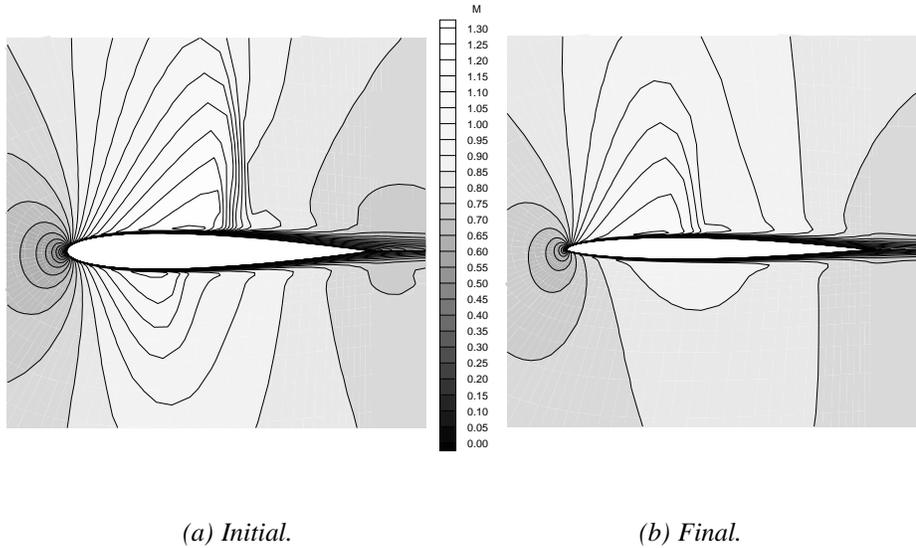
(a) Initial.   *(b) Final.*

Fig. 10. *Flow-field Mach number contour plots for turbulent transonic airfoil SAADO demonstration.*
analysis iterations or extended the Newton-like methods to include the iteration equation that guides the change of design variables. Ghattas and Orozco [28] and Ta'asan et al. [29] also developed procedures for SAND. Ghattas and Orozco derived an equation that relates the design changes to the changes in the flow solution, based on the sparsity of the Hessian matrix in the design optimization problem formulation. Ta'asan et al. integrated the aerodynamic design optimization within the multigrid method for aerodynamic analysis. The error reduction in each grid improves not only the quality of the flow solution but also the design improvement at that scale. Results from other SAND procedures are given in several papers also contained in this volume.

The point here is that SAND procedures produce aerodynamic designs much more efficiently than the NAND procedures. Solution times for the SAND procedures are measured in one to a few single-analysis times, whereas those for even very efficient NAND procedures are measured in many equivalent single-analysis times. This difference was shown for the present nozzle example discussed in section 2.2.1, where the SAND solutions varied from approximately 2 to 9 equivalent single-analysis times and the NAND solutions were another factor of 2 to 3 times larger. For the turbulent transonic airfoil examples discussed in sections 2.1.2 (NAND) and 2.2.2 (SAND), this difference is not as clear because structural wing-box thickness constraints were imposed only in the latter example and the computational grids were different. In addition, the SAADO was implemented with ADIFOR (forward-mode) direct differentiation rather than the (reverse-mode) more efficient adjoint formulation. Nevertheless, the efficient NAND procedure required approximately 36 equivalent single-analysis solution times; the SAND procedure required 22 single-analysis times. Based on Euler solution comparisons, the estimated time for a single-adjoint SAND procedure for the Navier-Stokes equations would be equivalent to 7 single-analysis times.

3 CFD Multipoint Design Optimization Plans

Multipoint design for any single-discipline optimization generally exhibits several facets of an MDO formulation. Features of the solutions at the multiple design points may appear as constraints or objectives (or parts of them) for the multipoint problem and must be jointly optimized and/or coordinated (which also may be posed as an optimization) in the space of multipoint design variables. This set of multipoint design variables may also contain subsets of variables for the different single-point problems. Thus, solutions at the multiple points can be viewed and treated as different disciplines. When some or each of the multiple design points involve a local design or determine a subset of the multipoint design variables, then multilevel optimization arises naturally. Arbitrary decompositions of the multipoint design problem may be based on lower-level partitionings guided by other considerations such as problem size, computer resources and codes, domain decomposition, etc.; such decompositions lead to partition interface conditions which are mismatched and must be rectified by an upper-level coordination.

Herein, coordination tasks are viewed as those involved in establishing multipoint (or multidisciplinary) feasibility; that is, upper-level constraints which establish compatibility of shared design variables, target values, and partition interface conditions for the multiple single-point analyses or sublevel optimizations. The number of levels of optimization or coordination provide another method for viewing multipoint design processes. A detailed description of design variables and objective and constraint functions at each optimization level depends, of course, on the particular decomposition of a given multipoint design problem. Our intention here is to show the multipoint nature of the MDO problem and suggest the need for efficient discipline codes for both analysis and design in order to solve the MDO problem efficiently, without specification of the decomposition details. Formal SA may or may not be implemented along with the iterative analyses; it is not indicated in the diagrams shown in this section. In the following subsections, general ideas related to planned CFD multipoint design optimization studies are used to illustrate the points to be made.

3.1 Single-Level Optimization

In the single-level processes, both optimization and coordination (OC) tasks are performed at the same level. The conventional, or NAND procedure, places these OC tasks above or in an outer iterative loop around the multipoint analyses, as shown in Fig. 11(a). The OC tasks provide design variables as input to the multipoint analyses, which upon inner iterative convergence provide the output functions (and perhaps the SD information) required by the OC tasks. A design variable update then initiates the next outer loop iteration. This procedure requires well-converged analyses at each of the multiple points for every change of design variables or iteration step in the outer (design) loop.

The single-level simultaneous, or SAND procedure, can be viewed either as incorporating the OC tasks within the multiple point iterative analysis loops or as running a modified NAND procedure with nonconverged inner or analysis loops. Both design and state variables are simultaneously updated, and the multiple point analyses are not converged until the desired design is obtained. As indicated in Fig. 11(b), both state and adjoint (costate or derivative) equations are iterated together, which simultaneously improves all multiple-point flow residuals and OC task objectives.
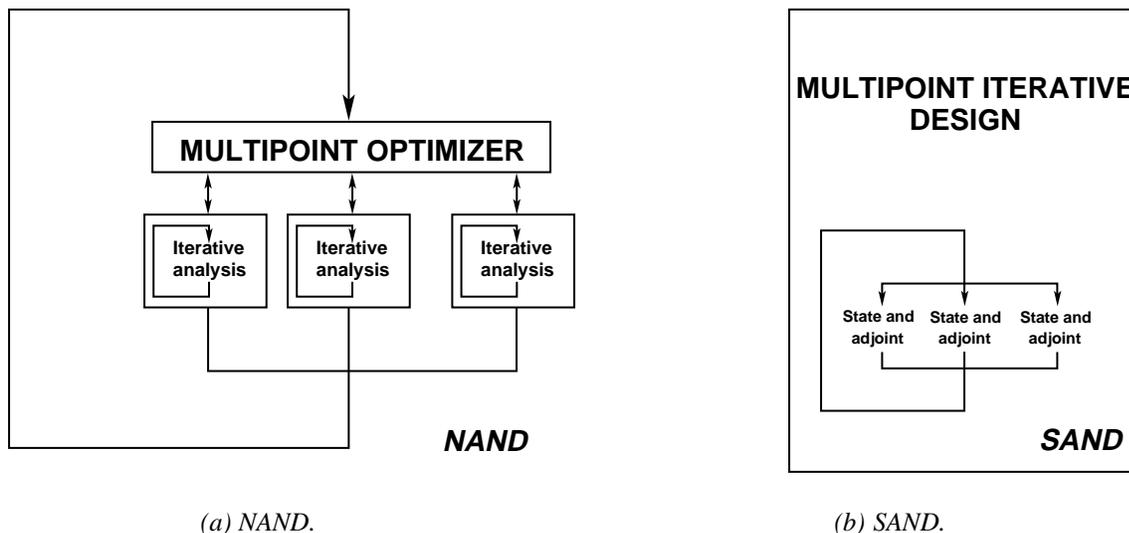

(a) NAND.   *(b) SAND.*

Fig. 11. *Multipoint, single-level optimization with iteratively solved state equations.*

3.2 Multilevel Optimization

In the multilevel processes, optimization and coordination tasks need not be performed at the same level; suboptimizations may be done at a lower or inner iteration level. Subsets of design variables that have been essentially determined at one design point may not be needed or relevant at other design points, or they may only be required to be not incompatible. For example, many design variables for the high-lift system are determined at takeoff and landing conditions but are not relevant at cruise conditions, where

11

efficient flight conditions drive the detailed aerodynamic shape design. Design variables which appear in more than one suboptimization must be subject to coordination, generally at a higher level. When more than one level of optimization exists, the procedure at each level is specified. A NAND-NAND procedure that involves CFD appears too expensive to be considered.

A multipoint, multilevel SAND-NAND procedure is pictured in Fig. 12 for three sublevel design points. The coordination task is assumed to be in the top-level multipoint SAND optimizer. The iterative analysis at each multiple design point is the innermost of three nested loops and may be called many times per optimization cycle, particularly if SA is not used to provide SD information. Repeated use of the iterative analysis has a dire effect when that iterative analysis is expensive CFD. Design variables at the top-level multipoint SAND optimizer generally include target values or parameters in the sublevel optimization objective functions. Aerodynamic shape optimization for the configuration or parts of it at low subsonic, transonic, and supersonic flow conditions might be the three sublevel design problems and the resulting shapes must be made compatible by the upper-level coordination at convergence in the outer loop.
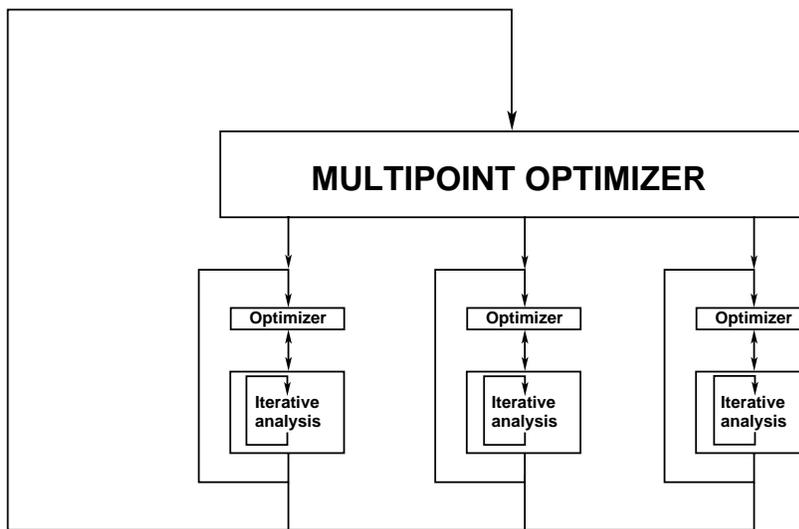

Fig. 12. *Multipoint, multilevel SAND-NAND optimization with iteratively solved state equations.*

A multipoint, multilevel SAND-SAND procedure is pictured in Fig. 13 for the same three sublevel design points as depicted in Fig. 12. Again, the coordination task is assumed to be in the top-level multipoint SAND optimization, where design variables include target values or parameters for the sublevel
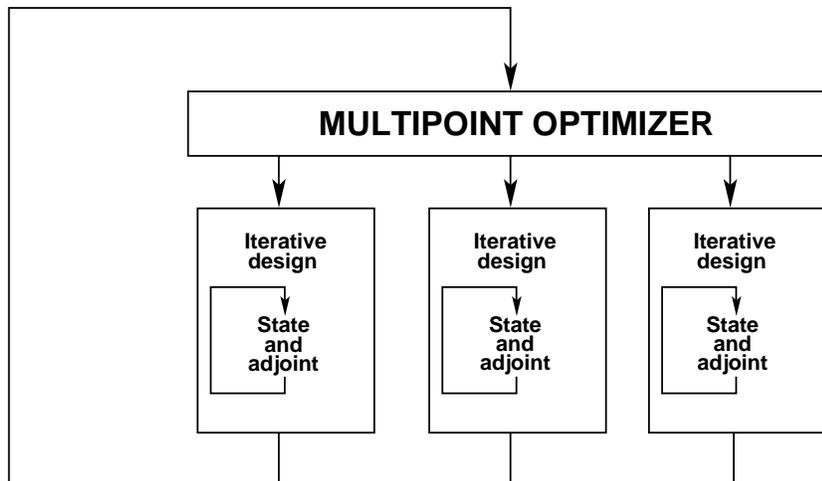

Fig. 13. *Multipoint, multilevel SAND-SAND optimization with iteratively solved state equations.*

objective functions. Use of SAND procedures for the sublevel iterative designs should be computationally more efficient than the corresponding NAND procedures with SA at the sublevel.

3.3 Hybrid Optimization

A combination of the single-level and multilevel processes will be called the hybrid optimization process. In Fig. 14, the multilevel part is pictured as a SAND-SAND procedure, whereas the single-level part is shown as a NAND procedure which may or may not include SA. Again, the coordination is assumed to be at the top-level multipoint optimization, or in the outer loop shown in Fig. 14. In a multipoint aerodynamic wing design that utilizes CFD techniques, for example, the three sublevel iterative design points (at the left) may be viewed as corresponding to takeoff and landing (low speed), transonic cruise, and supersonic cruise, as previously discussed. The iterative analyses (at the right) may correspond to other flow conditions (i.e., aerodynamic "off-design" points) where, for example, wing root bending moment, shock strength, lift, or drag (or other functions) need to be assessed at each multipoint design step for constraint evaluation or other considerations. Often, such constraints arise from other discipline considerations and are, therefore, required in a realistic single-discipline optimization. Updated input at these off-design analysis points would be determined by the coordination at the upper level multipoint design step.
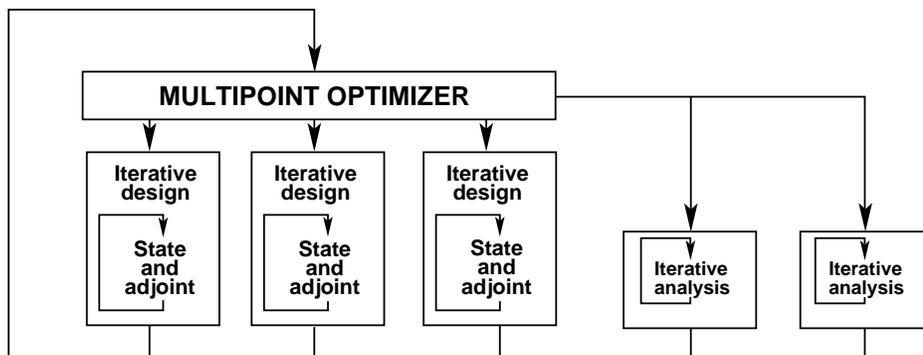

Fig. 14. *Multipoint, multilevel hybrid optimization with iteratively solved state equations.*

4 Suggested MDO Procedure

A given discipline multipoint design, however, differs from that shown in Fig. 14 because many of the required simultaneous analyses will be from other disciplines and will generally occur at the off-design points of those disciplines. For example, in the multipoint aerodynamic wing design pictured in Fig. 15, structural analyses will be required at each sublevel aerodynamic design point for a flexible wing and also
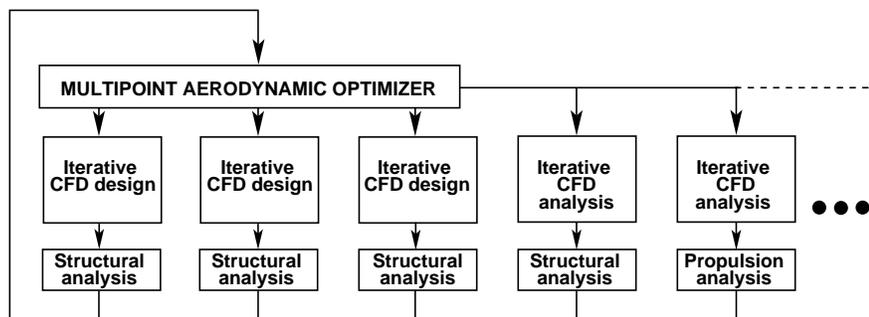

Fig. 15. *Multipoint, multilevel hybrid aerodynamic optimization with MDA.*

(possibly) at the aerodynamic off-design points. None of these points may be those that correspond to the structural design points. In addition, other discipline analyses may also be required for the multipoint aerodynamic optimizer. The same is true for the corresponding structural multipoint design; aerodynamic analyses are required at the "load-case" flow conditions, as shown in Fig. 16. Generally, these points

are not the aerodynamic (shape) design points. Thus, the set of multipoint MDO design points appears to be a collection of hybrid multilevel "discipline-design" multidisciplinary analysis (MDA) procedures, each with a different subset of design points or flow conditions. A practicable implementation of these procedures requires efficient computational code for both the discipline analysis and the iterative design. For the expensive iterative analyses of advanced CFD, the iterative design should be a SAND procedure rather than a NAND procedure with SA.
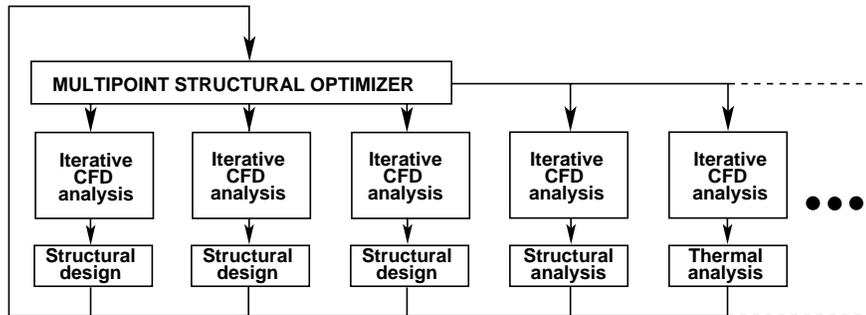

Fig. 16. *Multipoint, multilevel hybrid structural optimization with MDA.*

The suggested MDO formulation is a coordinated combination of hybrid multilevel processes such as those pictured in Figs. 15 and 16. The multipoint hybrid multilevel optimizations will be sublevel optimizations in an MDO coordination level as shown in Fig. 17. This suggested MDO formulation
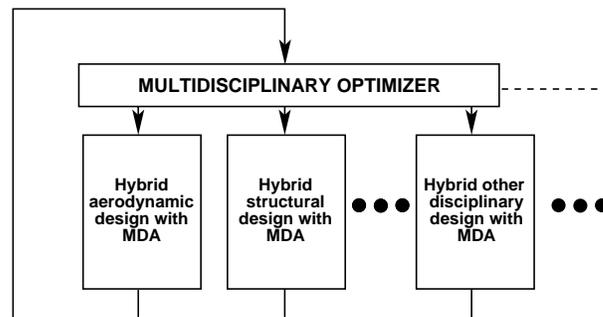

Fig. 17. *Suggested MDO procedure: coordinated hybrid multilevel suboptimizations.*

is much closer to the currently practiced design procedures than an "all at once" MDO procedure as described, for example, in [11]. In effect, each discipline actively participates in its sublevel hybrid design, with supporting MDA and MDO coordination from the top level or the outer loop. This suggested MDO formulation can also be viewed as a multilevel coordinated or a collaborative suboptimization procedure.

5 Concluding Remarks

Observations have been made regarding the use of advanced computational fluid dynamics (CFD) analysis, sensitivity analysis (SA), and design codes in gradient-based multidisciplinary design optimization (MDO). These observations reflect our perception of the interactions required of CFD and our experience in recent aerodynamic design optimization studies using CFD. Sample results from these latter studies have been summarized for conventional optimization (analysis-SA codes) and simultaneous analysis and design optimization (design code) using both Euler and Navier-Stokes flow approximations. The amount of computational resources required for aerodynamic design using CFD via analysis-SA codes is shown to be greater than that required via design codes. Thus, an MDO formulation that utilizes the more efficient design codes where possible is desired. However, in the aerovehicle MDO problem, the various disciplines that are involved have different design points in the flight envelope; therefore, CFD analysis-SA codes are required at the aerodynamic "off-design" points. It is concluded that the appropriate MDO formulation

14

include hybrid multilevel optimization procedures that consist of both multipoint CFD analysis-SA codes and multipoint CFD design codes that perform suboptimizations.

References

[1] P. A. Newman, G. J.-W. Hou, H. E. Jones, A. C. Taylor, III, and V. M. Korivi, *Observations on computational methodologies for use in large-scale, gradient-based, multidisciplinary design*, A Collection of Technical Papers, 4th AIAA/USAF/NASA/OAI Symposium on Multidisciplinary Analysis and Optimization, AIAA, Sept. 1992, pp. 531–542 (AIAA 92–4753–CP).

[2] A. C. Taylor, III, P. A. Newman, G. J.-W. Hou, and H. E. Jones, H. E., *Recent advances in steady compressible aerodynamic sensitivity analysis*, (Gunzberger, M. D., ed.), Flow Control, Volumes in Mathematics and Its Applications (IMA), 68, Springer-Verlag, 1995, pp. 341–356.

[3] C. Bischof, G. Corliss, L. Green, A. Griewank, K. Haigler, and P. Newman, *Automatic differentiation of advanced CFD codes for multidisciplinary design*, Computing Systems in Engineering, 3 (6), (1992), pp. 625–637.

[4] P. A. Newman, *Preparation of advanced CFD codes for use in sensitivity analyses and multidisciplinary design optimization*, (Borggaard, Burkardt, Gunzberger, and Peterson, eds.), Optimal Design and Control, Birkhauser, 1995, pp. 241–274.

[5] A. Carle, L. L. Green, C. H. Bischof, and P. A. Newman, *Applications of automatic differentiation in CFD*, AIAA Paper 94–2197, June 1994.

[6] L. Sherman, A. Taylor, L. Green, P. Newman, G. Hou, and M. Korivi, *First- and second-order aerodynamic sensitivity derivatives via automatic differentiation with incremental iterative methods*, A Collection of Technical Papers, 5th AIAA/NASA/USAF/ISSMO Symposium on Multidisciplinary Analysis and Optimization, AIAA, Sept. 1994, pp. 87–120 (AIAA 94–4262–CP).

[7] C. H. Bischof, A. Carle, G. Corliss, A. Griewank, and P. Hovland, *ADIFOR: Generating derivative codes from Fortran programs*, Scientific Programming, 1 (1), (1992), pp. 11–29.

[8] C. H. Bischof and A. Griewank, *ADIFOR: A Fortran system for portable automatic differentiation*, A Collection of Technical Papers, 4th AIAA/USAF/NASA/OAI Symposium on Multidisciplinary Analysis and Optimization, AIAA, Sept. 1992, pp. 433–441 (AIAA 92–4744–CP).

[9] G. J.-W. Hou, V. Maroju, A. C. Taylor, III, V. M. Korivi, and P. A. Newman, *Transonic turbulent airfoil design optimization using automatic differentiation in incremental iterative forms*, A Collection of Technical Papers, 12th AIAA Computational Fluid Dynamics Conference, AIAA, June 1995, pp. 512–526 (AIAA 95–1692–CP).

[10] R. J. Balling and J. S. Sobieski, *Optimization of coupled systems: A critical overview of approaches*, A Collection of Technical Papers, 5th AIAA/USAF/NASA/ISSMO Symposium on Multidisciplinary Analysis and Optimization, AIAA, Sept. 1994, pp. 753–773 (AIAA 94–4330–CP).

[11] G. R. Shubin, *Application of alternative multidisciplinary optimization formulations to a model problem for static aeroelasticity*, Journal of Computational Physics, 118 (1995), pp. 73–85.

[12] V. M. Korivi, P. A. Newman, and A. C. Taylor, III, *Aerodynamic optimization studies using a 3–D supersonic Euler code with efficient calculation of sensitivity derivatives*, A Collection of Technical Papers, 5th AIAA/NASA/USAF/ISSMO Symposium on Multidisciplinary Analysis and Optimization, AIAA, Sept. 1994, pp. 170–194 (AIAA 94–4270–CP).

[13] V. M. Korivi, A. C. Taylor, III, G. J.-W. Hou, P. A. Newman, and H. E. Jones, *Sensitivity derivatives for three-dimensional supersonic Euler code using incremental iterative strategy*, AIAA Journal, 32 (6), (1994), pp. 1319–1321.

[14] R. L, Barger and M. S. Adams, *Automatic computation of wing-fuselage intersection lines and fillet inserts with fixed-area constraint*, NASA TM–4406, Mar. 1993.

[15] R. L. Barger, M. S. Adams, and R. R. Krishnan, *Automatic computation of Euler marching grids and subsonic grids for wing-fuselage configurations*. NASA TM–4573, July 1994.

[16] G. N. Vanderplants, *ADS—A Fortran program for automated design synthesis*, NASA CR-17785, Sept. 1985.

[17] R. E. Smith, Jr. and M. R. Wiese, *Interactive algebraic grid-generation technique*, NASA TP-2533, Mar. 1986.

[18] G. Kuruvila, S. Ta'asan, and M. D. Salas, *Airfoil optimization by the one-shot method*, AGARD-FDP-VKI Special Course on Optimum Design Methods in Aerodynamics, Von Karman Institute, Rhode–St. Genese, Belgium, Apr. 1994.

[19] A. C. Taylor, III, *Convergence acceleration of upwind relaxation methods for the Navier-Stokes equations*, Ph.D. Dissertation, Virginia Polytechnic Institute and State University, Blacksburg, VA, July 1989.

[20] B. Baldwin and H. Lomax, *Thin-layer approximation and algebraic model for separated turbulent flows*, AIAA Paper 78–0257, Jan. 1978.

[21] G. W. Hou, A. C. Taylor, III, S. V. Mani, and P. A. Newman, *Simultaneous aerodynamic analysis and design optimization*, Abstracts from Second U.S. National Congress on Computational Mechanics, Aug. 1993, Washington, D.C., p. 130 (submitted for publication).

[22] S. V. Mani, *Simultaneous aerodynamic analysis and design optimization*, M.S. Thesis, Old Dominion University, Norfolk, VA, Dec. 1993.

[23] G. J.-W. Hou, V. M. Korivi, A. C. Taylor, III; V. Maroju, and P. A. Newman,, *Simultaneous aerodynamic analysis and design optimization (SAADO) of turbulent transonic airfoil using a Navier-Stokes code with automatic differentiation (ADIFOR)*, NASA Computational AeroSciences Workshop, Mar. 1995 (abstracts to be published as NASA CP).

[24] M. Rizk, *The single-cycle scheme: A new approach to numerical optimization*, AIAA Journal, 21 (1983), pp. 1640–1647.

[25] R. L. Campbell, *An approach for constrained aerodynamic design with application to airfoils*, NASA TP–3260, Nov. 1992.

[26] M. Drela, *Viscous and inviscid inverse schemes using Newton's method*, AGARD Report No. 780, 1990, pp. 9–1—9–16.

[27] D. P. Young, W. P. Huffman, R. G. Melvin, M. B. Bieterman, C. L. Hilmes, and F. T. Johnson, *Inexactness and global convergence in design optimization*, Presented at the 5th AIAA/NASA/USAF/ISSMO Symposium on Multidisciplinary Analysis and Optimization, AIAA, Sept. 1994, (AIAA 94–4386).

[28] O. N. Ghattas and C. E. Orozco, *Sparse approach to simultaneous analysis and design of geometrically nonlinear structures*, AIAA Journal, 30 (1992), pp. 1877–1885.

[29] S. Ta'asan, G. Kuruvila, and M. D. Salas, *Aerodynamic design and optimization in one shot*, AIAA Paper 92–0025, Jan. 1992.