NASA CR- 166,471

NASA CONTRACTOR REPORT 166497

IMPROVED METHOD FOR TRANSONIC AIRFOIL DESIGN-BY-OPTIMIZATION

Robert A. Kennelly, Jr.
Informatics General Corporation

Prepared for
Ames Research Center
under Contract NAS2-11555

National Aeronautics and
Space Administration

Ames Research Center
Moffett Field, California 94035

IMPROVED METHOD FOR TRANSONIC AIRFOIL DESIGN-BY-OPTIMIZATION*

Robert A. Kennelly, Jr.+
Informatics General Corporation
Palo Alto, California

Abstract

An improved method for use of optimization techniques in transonic airfoil design is demonstrated. FLO6QNM incorporates a modified quasi-Newton optimization package, and is shown to be more reliable and efficient than the method developed previously at NASA-Ames, which used the COPES/CONMIN optimization program. The design codes are compared on a series of test cases with known solutions, and the effects of problem scaling, proximity of initial point to solution, and objective function precision are studied. In contrast to the older method, well-converged solutions are shown to be attainable in the context of engineering design using computational fluid dynamics tools, a new result. The improvements are due to better performance by the optimization routine and to the use of problem-adaptive finite difference step sizes for gradient evaluation.

Introduction

Prediction of system behavior is often just a first step - the real goal is an improved system, or better yet, one that is optimal in some sense. Numerical optimization techniques provide one means for direct use of analysis methods in engineering design. A scalar objective function is specified such that a decrease in value due to changes in the design variables corresponds to an improvement. If a minimum can be found, it represents the "best" that can be achieved, at least locally. Optimization in this context only means solution of the formal problem posed; the suitability of the design obtained depends on the problem and the properties of the analysis program used. Mathematical programming techniques permit these problems to be solved convincingly - a point can be found where the sensitivity of the objective to the design variables (the gradient) is very much reduced from its value at the initial point, and where the change in the variables each design iteration is small. Specialized techniques, if available, may be more efficient, but optimization methods offer the advantages of generality and flexibility since many analysis methods can be adapted to this use and applied to a wide range of problems. Careful integration of analysis and optimization codes is required, however, and special attention must be paid to the robustness of the individual components for the sake of overall reliability.

The general aerodynamic design problem involves an objective function, and possibly constraints, formed from configuration geometry, computed pressures, forces, etc. The variables may include parameters describing the geometry or onset flow. Several design points defined by combinations of Mach number and angle of attack may be of interest, and if broader considerations such as structures or performance are to be taken into account, the problem formulation may be even more complex. An interesting special case is the "pseudo-inverse" problem where a pressure distribution is specified on a surface and it is desired to find a shape, or shape plus flow conditions, which will produce those pressures.

A number of authors have described methods and results of applications in subsonic, transonic, and supersonic flow using a variety of techniques for both aerodynamic analysis and optimization [References 1-11; numbers in brackets refer to bibliographic citations listed below]. The present work is an attempt to evaluate two design programs for robustness and efficiency. We are primarily concerned here with the relative performance of the methods, not with the designs obtained. The test cases consist of relatively simple two-dimensional transonic pseudo-inverse problems which are convenient because they can be constructed with known solutions, yet retain many features of more realistic applications. Subsonic problems were not considered, but this was not because of any limitation of the design codes.

From an optimization standpoint, significant problem characteristics include the following:

1. Objective function evaluations dominate the cost of a run
2. Only function values are available - derivatives with respect to the design variables must be approximated by finite differences
3. The number of variables is not so large that storage considerations need be considered in the choice of a method
4. The iterative flow solution technique allows the precision of the objective to be traded off against cost
5. The optimum point need not be determined to high precision since the final values of the individual variables are not of interest
6. The overall design process is fairly long, so that some user intervention is anticipated
7. The objective function may be occasionally discontinuous

---

* This paper is based on NASA CR- prepared for Ames Research Center under NASA Contract NAS2-11555.

+ Member of the Professional Staff, Member AIAA.

In this context, three interrelated goals have been pursued. First, demonstration of superior performance by a new technique on aerodynamically interesting test problems with known solutions. Second, the procedure was intended to illustrate the use of the new method and to provide examples of typical difficulties and their resolution. Finally, the work to be described may be helpful as a case study in the development of
design-by-
optimiz
istics noted
r many other
r the design

add copy

obtained
analysis
ient-type
codes are
.06 with
mization
re the
vith a
'. The
y Gill,
h was
evised
ied to
n the
ng of
e for
sizes
been
is
cks,
The
are

d

on a
difference mesh. A
mapping is employed to transform the flow region to the interior of a circle. Corrections to the discretized potential are computed by a fast Poisson solver combined with relaxation sweeps which are repeated one or more times per iteration. The analysis code was modified in several ways including provision for tightening some internal tolerances, revising storage allocations, and performing some detail changes for the sake of execution speed. The process of preparing FLO6 for coupling with the new optimization methods resulted in a number of improvements over the version used in FLO6OPT: it is now more reliable, permits better user control over the precision of the results, is easily modified to use different mesh densities, and is about 25% cheaper to run. (This cost comparison is based on work done on the NASA-Ames CDC 7600 computer, but the bulk of the results to be reported were obtained with the recently-acquired Cray-1S. No modifications to take advantage of the Cray's vector capabilities have yet been performed, so absolute program timings will not be emphasized in what follows.)

Convergence difficulties were encountered during preliminary testing, and several FLO6 parameters were varied in an effort to understand and cure the problem. Increasing the number of Fourier coefficients used in the mapping seemed to help one otherwise unremarkable case converge from a stored solution. (There was no problem obtaining a converged solution with the same case when starting from scratch.) The number of coefficients was left at the higher level, and the test cases proceeded without further incident. Subsequent computational experience has led to the conjecture that many convergence failures, discontinuities in optimization objective, and instances of a restarted solution not matching its progenitor are due to non-uniqueness in the flow solution itself. If this is correct, then the sensitivity of the computations to seemingly irrelevant factors such as overspecifying the number of Fourier coefficients is less surprising.

COPES/CONMIN Package

FLOSCPS makes use of a more recent version of CONMIN than did FLO6OPT, and uses the COPES front end program for input and control. The optimization package was originally written for structural design problems with large numbers of variables and constraints, but includes provision for computing unconstrained search directions when no constraints are active. Since both FLO6OPT and its three-dimensional sibling FLO22OPT have often been used in pseudo-inverse mode without constraints [10, 22-25], the unconstrained portion of the optimization code has acquired more significance in application to aerodynamic problems than it had in the original design. The Fletcher-Reeves conjugate gradient algorithm implemented for this purpose appears to have attractive theoretical properties, has low storage overhead, and is easy to implement in its basic form. It is not, however, the method of choice except for very large problems, where it may be the only alternative [21, 26]. At each iteration of this technique, the gradient is calculated and a line search performed along a ray given by a certain linear combination of the current gradient and the last search direction. Computer storage requirements are low because only a few vectors of length n need be accommodated, where n is the number of design variables.

Objective function gradients are calculated by forward differencing, with no provision for use of central differences. The step size employed can be scaled by the size of the associated variable, but is independent of the value of the objective function, its precision, or its derivatives.

FLO6OPT and FLO22OPT have not been entirely satisfactory in applications, and some users [22, 27] have resorted to use of only a few design variables at a time. This is inconvenient and can result in failure to solve even simple problems, as discussed in [26, 28-29], for example. The COPES/CONMIN package has no provision for restarting an interrupted run with data from a preceding one. This loss of history can lead to inefficiency since a restart must begin from scratch with the notoriously slow steepest descent search direction.

Quasi-Newton Method

A new program, FLO6QNM, has been developed for unconstrained design problems, or for problems whose few, loose constraints permit efficient use of penalty functions. It was hoped that an optimization algorithm more specifically suited to such problems would prove superior to COPES/CONMIN. QNMDIF is efficient, reasonably up-to-date and well documented, and contains safeguards to help maintain progress in the face of difficulties such as roundoff errors, ill-conditioning, or occasional small discontinuities in the objective function. It uses finite difference approximations to the derivatives of the objective with respect to the design variables.

The quasi-Newton algorithm, like Newton's method itself, attempts at each iteration k to step to the minimum of a quadratic model of the actual objective. This step, denoted p(k) below, is determined by the linear system

$$G(k)\, p(k) = -g(k)$$

where g(k) is the gradient and G(k) is, in Newton's method, the matrix of mixed partial second derivatives. In practice, a one-dimensional search is performed to locate a minimum along the ray p from the current point. Instead of G, which may be difficult or expensive to evaluate, quasi-Newton methods use a sequence of matrices B(k) which approximate G(k) in a certain sense. After the new point x(k+1) has been found and the gradient reevaluated, B is modified so that the quasi-Newton condition

$$[g(k+1) - g(k)] = B(k+1)\, [x(k+1) - x(k)]$$

holds, as it would if the objective were quadratic and if B were the true Hessian. This relationship is just the truncated Taylor series for the gradient about the point x(k). Using a rank-2 update to B, the approximate Hessian's symmetry and positive definiteness can be preserved, even in the presence of roundoff errors [19]. On a new problem, B(0) is usually taken to be the identity, but in FLO6QNM estimates of the diagonal elements are available if step size procedure FDSTEP has been used.

Theoretically, in the special case of a quadratic objective in n dimensions, the quasi-Newton algorithm converges to the minimum in n iterations, as should the algorithm used by CONMIN. However, the quasi-Newton algorithm does not depend on exact line searches, and indeed is usually more efficient when a relatively coarse line search is used [18]. This is an advantage over CONMIN because the conjugate gradient algorithm depends on exact searches for its good theoretical convergence properties. (CONMIN's search does not actually appear to perform exact searches anyway.)

Computational experience, and some analytical results, indicate that quasi-Newton methods converge superlinearly, or at least at a fast linear rate. Thus the asymptotic error constant, the limiting value of the error ratio for successive iterations, will be zero or at least small. In the convergence histories to be presented below, the difference between the current objective and its (known) optimal value is plotted against iteration number. Superlinear convergence is indicated by a curve which steepens as the optimum is approached. The appearance of this hoped-for behavior may be taken as supporting evidence that a true minimum point has been found [21].

Several features of QNMDIF are included for the sake of reliability. Since the approximate Hessian is positive definite by construction, QNMDIF generates descent directions at each step, providing the gradients are sufficiently accurate. If the line search fails to produce a significantly better point, the program switches to central difference gradients, and can switch back again later if an improved rate of progress warrants it. A further degree of adaptability is available: FLO6QNM can be restarted with tighter aerodynamic convergence tolerances if even better gradient information is required. The optimization routine has been modified to take account of objective function precision, and to permit resumption of an interrupted run without loss of information. Other enhancements include a user interface between QNMDIF and the analysis code which permits housekeeping chores such as printing FLO6 results after each optimization iteration, and provision for executing a local search at the beginning of a run as a means of moving away from a region in the design space where the objective is discontinuous.

For problems such as those under consideration here, with few variables and an objective with large storage and CPU time requirements, the additional overhead due to QNMDIF is negligible. (CPU refers to a computer's Central Processing Unit.)

Test Cases

The test cases consist of "reinventing" a representative supercritical airfoil using three different sets of design variables. The parameterizations were all chosen to be able to represent accurately the modifications required to transform the upper surface of a NACA 63A210 section into the target airfoil. Though superficially similar, the three problems appear very different to the optimizer. In each case, the perturbation to the initial surface is a linear combination of smooth functions, identically zero at the leading and trailing edges, and a single linear ramp term which allowed for non-zero trailing edge thickness. The various terms were normalized so that they all have a maximum value of about 1.0 over the airfoil's chord. The optimal coefficients of the linear combination representing the solution, and in one case the nonlinear parameters of the perturbing functions themselves, were determined by performing a general least squares fit to the difference between the known initial and final shapes. These three sections, consisting of initial airfoil plus fitted difference terms, were then analyzed using FLO5. For each case, pressure coefficients at sixty-two points from about 0.002 to 0.997 x/c along the upper surface of the target airfoil were extracted from the output of the analysis program to serve as targets for the design problems. Initial and target airfoils and Cp distributions are illustrated in Figure 1, where x/c is the normalized chordwise coordinate and Cp is the local static pressure minus the free stream static pressure, divided by the free stream dynamic pressure. The objective function for each problem is defined as the sum of the squared differences between the target and calculated pressure coefficients. The solution determined by the geometrical least squares minimization solves the aerodynamic problem too, since the non-negative objective is zero at this point. Note that this test procedure is more stringent than simply running the two design codes and comparing the results, since the possibility exists here for both to fail!

The run conditions for the analysis program were as follows:

1. Mach number 0.76

2. Angle of attack 0.0 degrees

3. 256 Fourier coefficients used in the mapping

4. Mapping tolerances 1.0E-10

5. Mesh size 128 circumferential x 32 radial

6. 8 (sometimes 6) relaxation sweeps per hybrid FLO6 iteration

7. Convergence tolerance for maximum residual 1.0E-8

(Note use of the FORTRAN convention for powers of ten: 1.0E-1 is one-tenth.) The sum-of-squares objective function was precise to about 1.0E-6 under these conditions, as estimated from values of the objective which were computed and printed out as the flow solution converged. Average CPU time per flow solution was about seven seconds on the Cray-1S, starting each solution from the potential array computed and stored at the beginning of every design iteration.

The level of precision chosen for these tests was a compromise. Six figures are more than adequate for the first case: usable results were obtained very quickly by FLO6QNM with a convergence criterion of 1.0E-4, which gives only about two good figures in the objective function! But as will be shown, the more difficult (and more realistic) problems require tighter tolerances. A single, intermediate degree of precision was chosen to illustrate the sort of results one might obtain without foreknowledge.

The available input parameters for the two optimization routines were different, complicating the attempt to compare them. Most inputs were simply left at their default values, but a few choices had to be made. Among these, the choice of step size for the gradient evaluations was most problematic. FLO6QNM permits automatic choice of optimal step lengths, where the cancellation error in the gradient due to the limited precision of the objective function is balanced against the truncation error resulting from the use of the finite difference approximation to the gradient. The default step size employed by COPES/CONMIN is 0.01 times the current value of each variable, but never less than 0.01. That scheme did not work well in initial trials. It was replaced by a uniform value of 0.001 for all of the variables, following the rule of thumb that the step size should be about the square root of the relative precision of the objective function [21]. This is nearly the optimal choice here in any event, since it is approximately the average of the values generated by the step length procedure in FLO6QNM for these well-scaled problems.

A multiplicative scaling factor was applied to each of the design variables. In the cases to be reported here, a factor of one hundred was sufficient to put all the first and second derivatives in the range 0.02 to 20. With this scaling, a design variable with value 1.0 changes the section ordinates by at most 0.01.

QNMDIF requires that the projected gradient along the direction of search be reduced to a specified fraction of its initial value. In design applications with expensive gradient evaluations, it was felt to be desirable to make the most of each iteration, so this fraction was set to 0.10 for a "moderate accuracy" search. In the cases to be described, the average number of steps per line search was between four and five.

It was not possible to compare the number of optimization steps required for "convergence" since that is ill-defined in this context of relatively crude function values and finite difference gradient approximations. Instead, the termination criteria were set tight enough to permit comparison of the rates of convergence over a reasonable number of steps. The comparison is not exhaustive, but is intended to demonstrate that the convergence rates predicted for quasi-Newton methods can be achieved in practice with "unfriendly" objective functions. The superiority of these methods over conjugate gradient techniques, as previously found in more elaborate comparisons using cheaper, more precise objectives, will be shown to hold true in the context of aerodynamic design. Reviews of computational experience regarding the relative robustness and efficiency of conjugate gradient and quasi-Newton methods may be found in [21, 26, 29-30].

<u>Test Case I</u>

The first test case was specially constructed to be easy to solve. The design variables were chosen to permit matching the target airfoil with only five "tuned" perturbing functions. Their general forms were as follows:

1. [1-(x/c)]*(x/c)**n / EXP [m*(x/c)]

2. SIN [pi*(x/c)**n]**m

where pi = 3.14..., and FORTRAN algebraic notation has been used. These are similar to the terms used in [9], but in addition to the coefficients of the linear combination, the parameters m and n controlling the width and location of the peak were systematically adjusted using a nonlinear least squares program to yield a good geometric fit. The shape parameters were then frozen at their optimal values and only the linear coefficients were used in the aerodynamic optimization. The final set of functions are shown in Figure 2.

Even in this simple case, the performance of the two design programs is different: in terms of both objective function value and norm of the final gradient, FL06QNM achieves a better result, in less time, than does FL06CPS. The results obtained are summarized in Table 1. FL06CPS performed 111 flow evaluations while FL06QNM required only 93 to obtain a significantly better result. The function count for FL06QNM, but not FL06CPS, includes extra objective evaluations for the step size calculations and for the estimation of the final gradient, necessary for checking convergence. Viewed another way, FL06QNM required only five iterations to reach the objective function level of FL06CPS after ten. But the goal of this work was to improve reliability as well as efficiency, and some additional runs were made to check this.

	FL06CPS	FL06QNM
Objective (initially 3.08)	0.00088	0.0000011
Gradient norm (initially 6.55)	0.11	0.0032
Optimization iterations	10	7

Table 1. Test Case I summary.

In addition to the basic case, a series of perturbed problems was considered. Arbitrary sets of design variables were deliberately perturbed by factors of ten within the analysis program in an effort to model the sort of uncertainty regarding problem scaling which is typical of many applications. The results with FL06CPS on this set of problems, using uniform finite difference step sizes, are presented in Figure 3. Of the eight problems attempted, only the base (unperturbed) run reached an objective value below 0.01 (which would represent a fairly good match to the target for this problem); the others are unsatisfactory. Next, the convergence histories obtained when FL06QNM was used with the same uniform set of step sizes are shown in Figure 4. While the new program solved all the problems, there is a fair degree of variation from run to run. Finally, with the addition of the automatic step size selection to FL06QNM, the results of Figure 5 were obtained. The curves have collapsed into one, with consistently rapid convergence for all versions of the problem. Thus even for mildly misscaled problems, use of good finite difference intervals for approximating the gradient can make a significant improvement in efficiency, especially when the available precision is limited. Alternatively, proper choice of step sizes may permit use of coarser (and perhaps cheaper) flow calculation tolerances. In summary, well-converged solutions were obtained by FL06QNM for all of the perturbed problems, including those where "incorrect" step sizes were used, while FL06CPS found only a mediocre solution to the unperturbed case and failed on the others.

Test Case II

The basis set for Test Case II consists of a "naive" collection of functions adapted from [10]. The form of the functions was the same as in the first test case, but in this more realistic problem no effort was made to adapt the basis set to the known solution. Although thirteen terms were used, the geometric fit to the target was not as good as for Test Case I. However, since the pressures computed from the fitted surface were used in forming the objective function, the optimal value of the objective was still near zero.

The perturbing functions consisted of two exponential terms, with peaks near the leading edge, sine terms centered at 5% and also at 10% intervals along the chord, and a linear ramp. The normalized shape functions are displayed in Figure 6. They are not as geometrically distinct as the special set derived for artificially simple Test Case I, and their similarity may help explain the slow convergence of both design codes shown in Figure 7. The upper curves show that when started from scratch, neither method converges with the rapidity exhibited in the previous case, although FL06QNM does again achieve a better result than FL06CPS. Several attempts to restart the FL06QNM run midway to calculate improved finite difference step sizes fared no better.

	FL06CPS	FL06QNM
Objective (initially 3.23)	0.0137	0.00510
Gradient norm (initially 8.2)	0.12	0.073
Optimization iterations	13	32

Table 2. Test Case II summary.

It seemed possible that both programs were converging to some local minimum other than the known one since despite the relatively small values of the objective, the profiles obtained did not match the target. What had been found was nearby points in the design space representing airfoils which supported pressures similar to the target - the true solution was quite a bit farther from the initial point. Several experiments were performed to see if the target could be reached. The lower pairs of curves in Figure 7 illustrate the effect of starting closer to the known solution. The base runs (upper curves) began with all variables set to zero. The middle set of curves result from starting at 90%, and the lower set at 99%, of the "correct" values. The FL06QNM runs converged promptly (but still not to the target) while FL06CPS showed the same behavior as in the base run - initial design improvement followed by leveling off and terminating due to lack of progress. Each run stopped at about the same distance from the solution (in the 13 dimensional design space) as it had begun. Evidently, a number of different points in the space gave rise to shapes which resembled the target airfoil fairly closely, and under the present conditions neither program was able to make the fine distinctions required to reach the known solution. This difficulty reflects the poorly conditioned geometric fit to the solution by the overlapping elements of the basis function set used. Further runs showed that FL06QNM could reach the true solution to the problem from the original starting point, but only by tightening the convergence criterion to provide additional precision, about 1.0E-8. Seventy design iterations were required for the objective to drop to 1.0E-6. It is reassuring that such problems can be solved if necessary, but this might be unreasonably expensive in a normal design context. Near the end of the run, rapid convergence similar to that obtained in Test Case I was displayed, as illustrated in Figure 8. At the final point of this long, composite calculation, which was restarted a number of times, the norm of the gradient had dropped to 3.0E-5 and the error in the design variables was less than 1% relative to the known solution, a much better result than obtained previously with cruder function values. The effects of objective function precision are explored more systematically in the final test case.

Test Case III

The basis set here consists of integrals of the so-called Wagner functions, described and first applied to airfoil optimization by Ramamoorthy and Padmavathi [7]. They resemble distorted sine functions, and are defined as follows:

1. [theta+SIN (theta)]/pi - [SIN (theta/2)]**2 (for n=0)

2. SIN [(n+1)*theta]/[pi*(n+1)] + [SIN (n*theta)]/pi (for n>0)

where theta = 2 * ASIN [SQRT (x/c)], and n is the series index. A sketch of the first nine members of the set is presented in Figure 9. These terms plus a linear ramp were sufficient to produce a very good match to the original target, both geometrically and aerodynamically. In fact, these ten basis functions permitted a better representation of the target than the thirteen used in Test Case II. (A Fourier sine series based on the appropriate periodic continuation of the airfoil upper surface also did not fit as well, even with several times as many terms.) In the base runs for this problem, with six digits of precision in the objective function, FL06CPS exhibited slow linear convergence, eventually reaching an objective value of about 0.01 after twenty iterations; the norm of the final gradient was 0.17. In contrast, FL06QNM quickly (in eight iterations) achieved an objective of 0.001 but then leveled off before terminating with final gradient norm 0.002.

	FL06CPS	FL06QNM
Objective (initially 3.19)	0.0118	0.00112
Gradient norm (initially 5.6)	0.17	0.0022
Optimization iterations	20	12

Table 3. Test Case III summary.

To investigate the plateauing behavior of FL06QNM here, and in Test Case II, additional runs were made with both design programs. The various convergence histories are compared in Figure 10, where the two sets of curves show the effect of objectives precise to roughly 1.0E-4, 1.0E-6 (the base case), and 1.0E-8. In the lower set, the plateau encountered by FL06QNM where the objective remained nearly constant for several iterations was followed by rapid convergence to a level of 1.0E-9 at iteration twenty-three with the benefit of more precision. FL06CPS, on the other hand, continued its slow convergence and reached only 0.005 (not even down to the level reached by FL06QNM with coarser tolerances) after twenty-five optimization steps. Final gradient norms were on the order of 0.2 and 0.0001 for FL06CPS and FL06QNM, respectively. A striking feature of the set of three FL06CPS trials is the absence of any effect on the convergence rate due to varying the objective precision. The slow progress is not improved when better information is provided. Increased precision is reflected in the better results obtained by FL06QNM at each level. Although the convergence of FL06QNM was not steady while distant from the optimum, the ultimate convergence rate was high and the final solution obtained was satisfactory.

The results of Test Case III were also used to obtain a comparison of the relative "costs" of the two methods. In Figure 11, the value of the objective is plotted against CPU time on the Cray-1S. The time intervals were estimated by multiplying the number of objective function evaluations for each iteration by the average time per evaluation. The cost per iteration is evidently about the same for FL06CPS and FL06QNM, but the progress after the first few iterations differs substantially, favoring the new method. The average cost per flow solution of the three Test Case III trials is given in Table 4. Note that the precise calculations required for the well-converged solutions are not too much more expensive than the looser ones used in the less successful runs.

Based on the number of iterations required for convergence on the fine mesh when starting from scratch, these results might have been expected to show a larger penalty for the tighter cases. For example, in the initial FL06 calculation for Test Case III, the maximum residual reached 1.0E-6 after seven flow iterations but took nineteen iterations to reach 1.0E-10 - about two and one-half times as long. The actual ratio is less than this, only about one and one-half, because the gradient is approximated using finite difference intervals which decrease roughly as the square root of the precision level. For the flow solutions required in the gradient calculations, the physical configuration is only changed a small amount. The stored potential array therefore provides a very good initial estimate, which improves as the convergence tolerance, and hence the corresponding finite difference intervals, are tightened. Since the gradient calculations take more than half of the total CPU time, this effect significantly mitigates the cost penalty of the more precise function values. In

Maximum residual	Precision	CPU time
1.0E-6	1.0E-4	5.2
1.0E-8	1.0E-6	7.0
1.0E-10	1.0E-8	8.0

Table 4. FL06 convergence criterion, objective precision, and average Cray-1S CPU time per flow calculation (in seconds) for Test Case III.

addition, progress per CPU second favors the run with the loosest tolerances only in the beginning, presumably because their better gradient information permits the tighter runs to catch up after a few, albeit more expensive, iterations.

Discussion

The main result of this work is that for pseudo-inverse transonic design, the new method is capable of really solving the optimization problems posed, in contrast to its precursor. FLO6QNM has also been shown to reach useful levels of the design objective function more efficiently than FLO6CPS. In each test case, and in each variation, this technique produced better designs more quickly, and it achieved greater reductions in the gradient of the objective function - enough to give good evidence of optimality. An additional indicator of true convergence was the high final convergence rates observed in the FLO6QNM runs, at least when sufficiently precise objective function values were available. As demonstrated with Test Case I, the new code is less sensitive to bad problem scaling, especially when the finite-difference step sizes are chosen appropriately.

The difficulties due to limited precision in the analysis program are surmountable, although not always cheaply. The step size selection method employed can be very useful, but requires two or more function evaluations per design variable at the beginning of a run to make its estimates. In FLO6QNM, this information can be used to provide initial estimates of the gradient and Hessian, thus recovering part of the cost involved. The switch to central differencing when progress is slow or when the gradient is small, another adaptive feature of QNMDIF which depends on objective precision, can also increase the cost of each iteration but may permit progress where other methods fail.

An important difference between the methods is that FLO6QNM is systematically able to take advantage of increased precision should it be necessary. The results of the third test case seem to indicate that if well-converged solutions are needed, there is an advantage to using tight tolerances from the beginning. (Just how tight depends on the problem at hand.) One part per ten thousand seems to be a minimum requirement, but will not be enough for more complex problems. In view of this uncertainty, the precision-adaptive and restarting capabilities of FLO6QNM are important advances over the more limited FLO6CPS.

The differences between the performances of COPES/CONMIN and QNMDIF, as modified for this application, amount to the difference between a design improvement method and an optimal design method. Since the posing and interpretation of aerodynamic design problems with current analytical tools is already difficult, the ability to solve such problems reliably is a significant step in the development of a systematic design methodology. The designer does not generally know whether a satisfactory solution to his problem exists at all. If the individual subproblems attacked can be solved for the best design that can be achieved at each stage, the overall process can proceed more surely. The alternative is to be forced to rely on guesswork to determine when to stop, tighten tolerances, restart from a different point, or reformulate the problem.

It may be worthwhile to reiterate that satisfactory results depend on careful preparation as well as choice of optimization method. The analysis code must be very reliable, produce consistently precise calculations, and must be fast, since a large number of analyses may be required for convergence to the desired level. Good programming practices make these requirements easier to meet, and may indeed be vital for proper control of what can become a fairly complex package. A degree of familiarity with the capabilities and limitations of the analysis method is necessary for realistic problem formulation. The best that an optimization method can do is to minimize efficiently the objective function with respect to the design variables provided - it is the designer's responsibility to ensure that the result will be a useful improvement in engineering terms.

Concluding Remarks

1. An improved airfoil design method has been demonstrated. In comparison testing on a series of transonic pseudo-inverse problems

with known solutions, FLO6QNM was both more reliable and more efficient than its predecessor. Clear evidence of convergence to the solution was obtained in every case.

2. The improvements are due to better performance by optimization code QNMDIF and to the use of problem adaptive finite difference step sizes.

3. The new program makes good use of available objective function precision. Its performance improves as tighter tolerances permit more accurate gradient estimates.

4. The observed differences between the quasi-Newton algorithm as implemented in QNMDIF and the conjugate gradient algorithm in COPES/CONMIN for unconstrained problems are consistent with the results of testing with simpler problems. When the analysis program FLO6 was modified to produce objective function values with a known, consistent level of precision, and the optimization scheme appropriately adapted to that level, superlinear or fast linear ultimate convergence rates were obtained with FLO6QNM on realistic aerodynamic test problems.

References

1. Hague, D. S., H. L. Rozendaal, and F. A. Woodward. "Application of Multivariable Search Techniques to Optimal Aerodynamic Shaping Problems." J. Astronaut. Sci. 15 (1968): 283-296.

2. Hicks, R. M., E. M. Murman and G. N. Vanderplaats. "An Assesment of Airfoil Design by Numerical Optimization." NASA TM X-3092, 1974.

3. Hicks, R. M., and G. N. Vanderplaats. "Application of Numerical Optimization to the Design of Low-Speed Airfoils." NASA TM X-3213, 1975.

4. Vanderplaats, G. N., R. M. Hicks, and E. M. Murman. "Application of Numerical Optimization Techniques to Airfoil Design." In Aerodynamic Analysis Requiring Advanced Computers, Part 2. NASA SP-347, (1975).

5. Hicks, R. M., G. N. Vanderplaats, E. M. Murman, and R. R. King. "Airfoil Section Drag Reduction at Transonic Speeds by Numerical Optimization." SAE Paper 760477, 1976.

6. Eggleston, B. and D. J. Jones. "The Design of Lifting Supercritical Airfoils Using a Numerical Optimization Method." Canadian Aeronautics and Space Journal 23 (1977): 172-181.

7. Ramamoorthy, P. and K. Padmavathi. "Airfoil Design by Optimization." J. Aircraft 14 (1977): 219-221.

8. Hicks, R. M., and P. A. Henne. "Wing Design by Numerical Optimization." AIAA Paper 77-1247, 1977.

9. Hicks, R. M. and G. N. Vanderplaats. "Application of Numerical Optimization to the Design of Supercritical Airfoils without Drag Creep." SAE Paper 770440, 1977.

10. Haney, H. P., R. R. Johnson, and R. M. Hicks. "Computational Optimization and Wind Tunnel Test of Transonic Wing Designs." AIAA Paper 79-0080, 1979.

11. Eggleston, B., D. J. Jones, and G. M. Elfstrom. "Development of Modern Airfoil Sections for High Subsonic Cruise Speeds." AIAA Paper 79-0687, 1979.

12. Jameson, A. "Transonic Flow Calculations for Airfoils and Bodies of Revolution." Grumman Aerospace Corporation Aerodynamics Report 390-71-1, n.d.

13. Jameson, A. "Iterative Solution of Transonic Flows over Airfoils and Wings, Including Flows at Mach 1." Comm. Pure Appl. Math. 27 (1974): 283-309.

14. Jameson, A. "Numerical Computation of Transonic Flows with Shock Waves." In Symposium Transsonicum II, edited by K. Oswatitsch and D. Rues. Berlin: Springer, 1976.

15. Jameson, A. "Transonic Flow Calculations." In Numerical Methods in Fluid Dynamics, edited by H. J. Wirz and J. J. Smolderen. Washington: Hemisphere, 1978.

16. Vanderplaats, G. N., and F. Moses. "Structural Optimization by Methods of Feasible Directions." Computers and Structures 3 (1973): 739-755.

17. Vanderplaats, G. N. "CONMIN - A FORTRAN Program for Constrained Function Minimization." NASA TM X-62,282, 1973.

18. Gill, P. E., W. Murray, and R. A. Pitfield. The Implementation of Two Revised Quasi-Newton Algorithms for Unconstrained Optimization. National Physical Laboratory, Division of Numerical Analysis and Computing Report No. NAC 11, 1972.

19. Gill, P. E., and W. Murray. "Quasi-Newton Methods for Unconstrained Optimization." J. Inst. Maths. Applics. 9 (1972): 91-108.

20. Wright, M. H. "Optimization." Computer Science Department, Stanford University, n.d. Photocopy.

21. Gill, P. E., W. Murray, and M. H. Wright. Practical Optimization. London: Academic Press, 1981.

22. Haney, H. P. "A-7 Transonic Wing Designs." In Transonic Aerodynamics, edited by D. Nixon. New York: American Institute of Aeronautics and Astronautics, 1981.

23. Bhateley, I. C., and R. A. Cox. "Application of Computational Methods to Transonic Wing Design." In Transonic Aerodynamics, edited by D. Nixon. New York: American Institute of Aeronautics and Astronautics, 1981.

24. Hicks, R. M. "Transonic Wing Design Using Potential-Flow Codes - Successes and Failures." SAE Paper 810565, 1981.

25. Lores, M. E., and B. L. Hinson. "Transonic Design Using Computational Aerodynamics." In Transonic Aerodynamics, edited by D. Nixon. New York: American Institute of Aeronautics and Astronautics, 1981.

26. Fletcher, R. Practical Methods of Optimization. Vol. 1, Unconstrained Optimization. Chichester: John Wiley and Sons, 1980.

27. Hinson, M. L. "A Series of Airfoils Designed by Transonic Drag Minimization for Gates Learjet Aircraft." In Transonic Aerodynamics, edited by D. Nixon. New York: American Institute of Aeronautics and Astronautics, 1981.

28. Swann, W. H. "Direct Search Methods." In Numerical Methods for Unconstrained Optimization, edited by W. Murray. London: Academic Press, 1972.

29. Himmelblau, D. M. Applied Nonlinear Programming. New York: McGraw-Hill, 1972.

30. Aoki, M. Introduction to Optimization Techniques. New York: Macmillan, 1971.

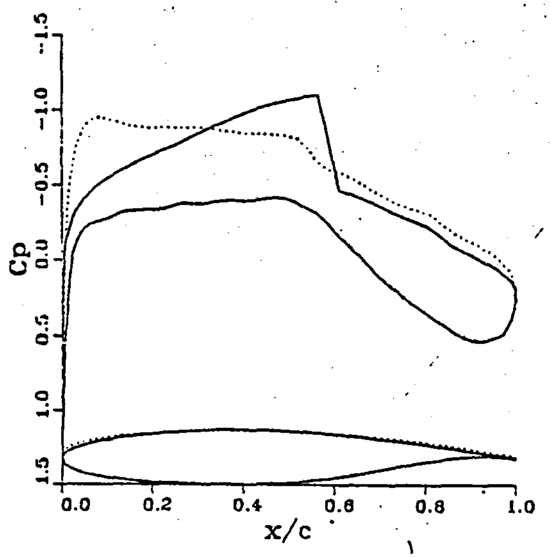

Figure 1. Sample test case Cp distributions and airfoils. Solid curves indicate initial configuration, dotted curves show target.

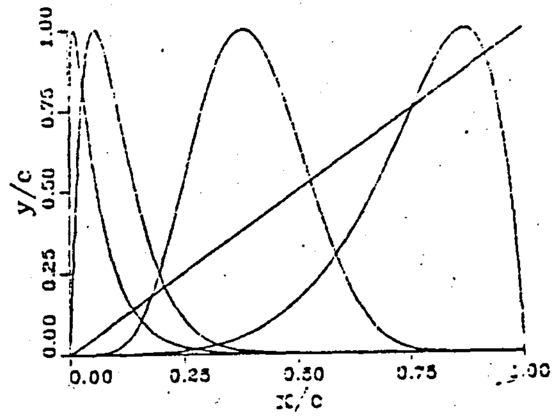

Figure 2. Test Case I shape functions. Exponential and sine type, plus a linear ramp.

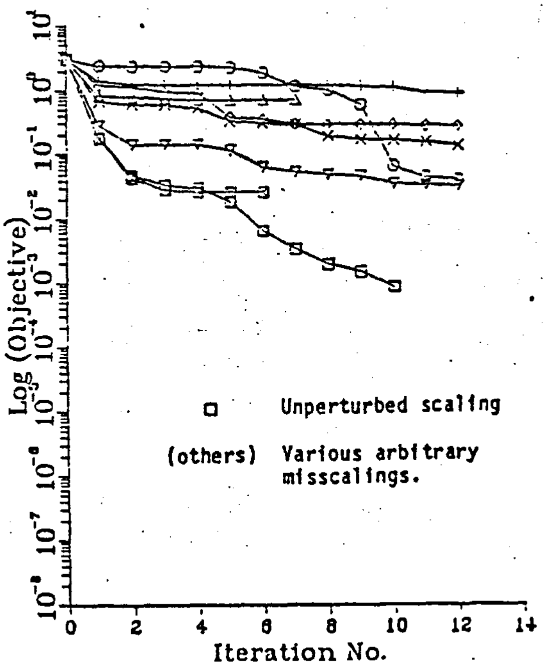

Figure 3. Test Case I iteration history. Program FLO6CPS with uniform finite difference steps of 0.001.

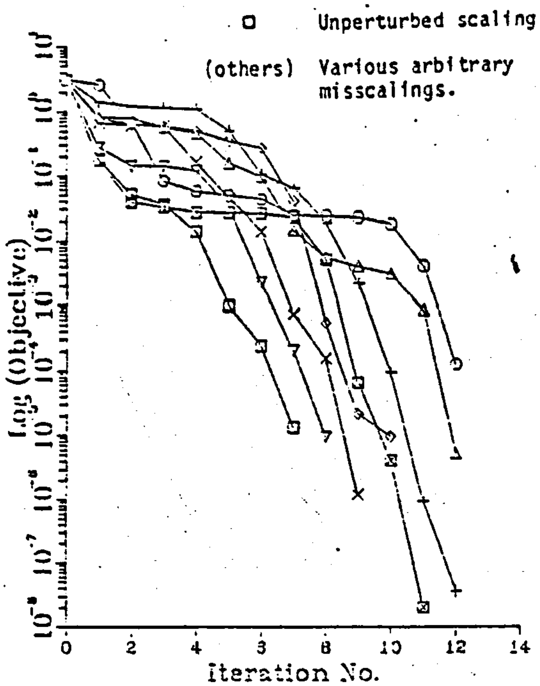

Figure 4. Test Case I iteration history. Program FLO6QNM with uniform finite difference steps of 0.001.

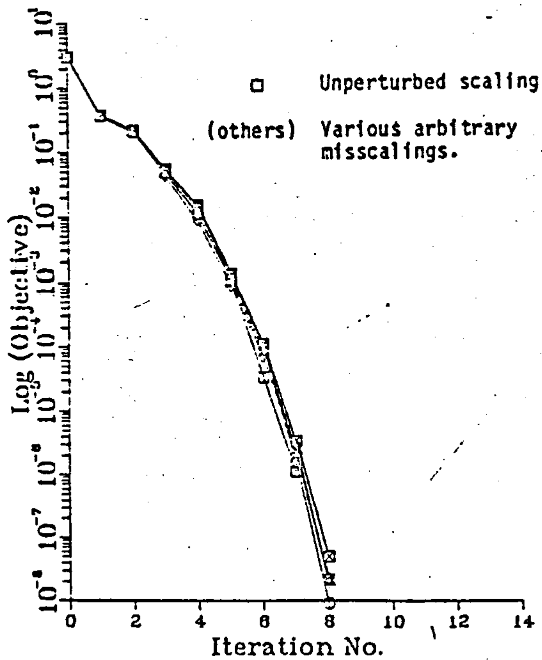

Figure 5. Test Case I iteration history. Program FLO5CPS with finite difference steps chosen by procedure FDSTEP.

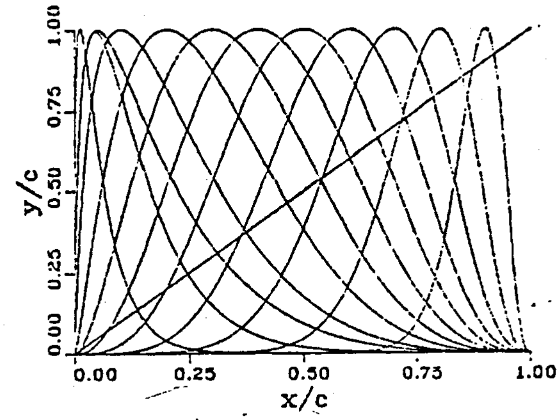

Figure 6. Test Case II shape functions. Exponential and sine type, plus a linear ramp.

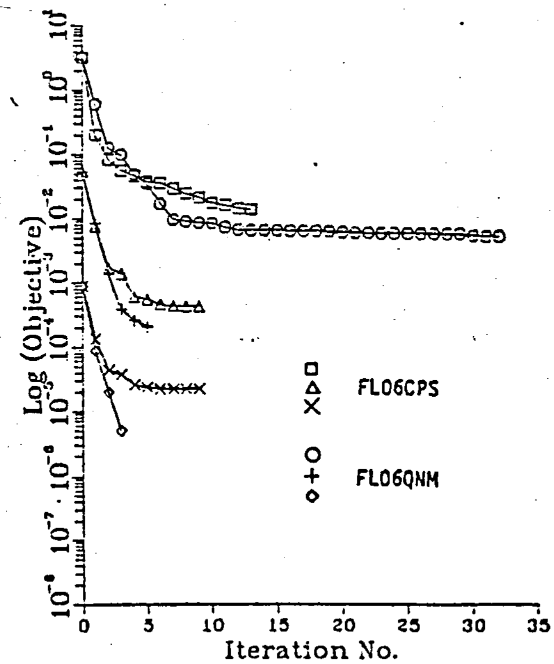

Figure 7. Test Case II iteration history. Upper curves represent "standard" initial point, middle and lower pairs result from starting at 90% and 99% of the solution, respectively.

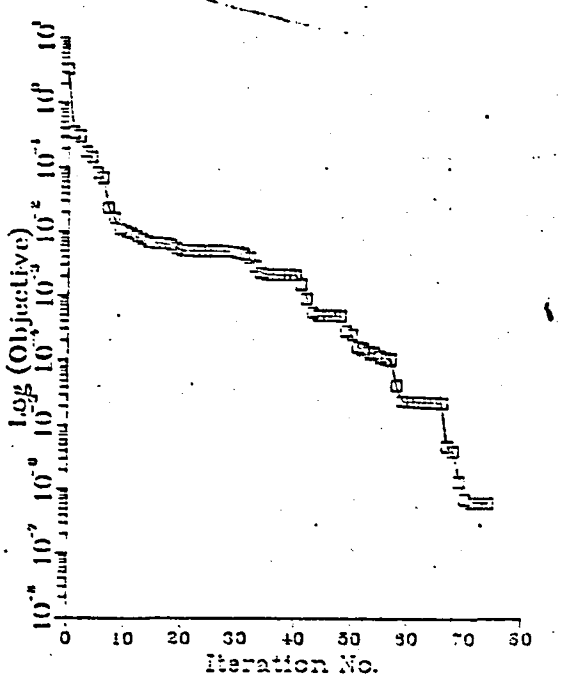

Figure 8. Special Test Case II iteration history. Program FLO6QNM with enhanced objective function precision.

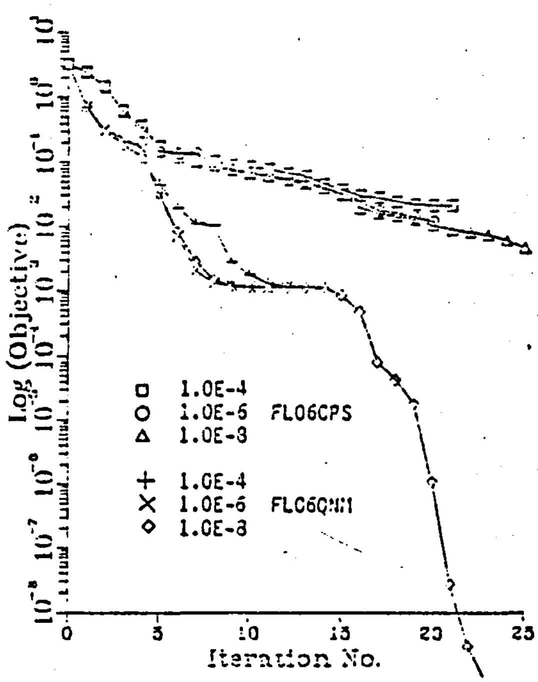

Figure 10. Test Case III iteration history. Effect of objective function precision on convergence rates of FLO6CPS and FLO6QNM.

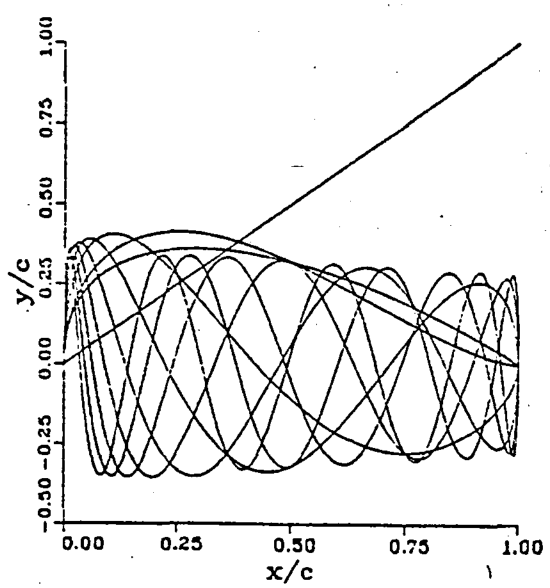

Figure 9. Test Case III shape functions. Wagner function integrals, plus a linear ramp.

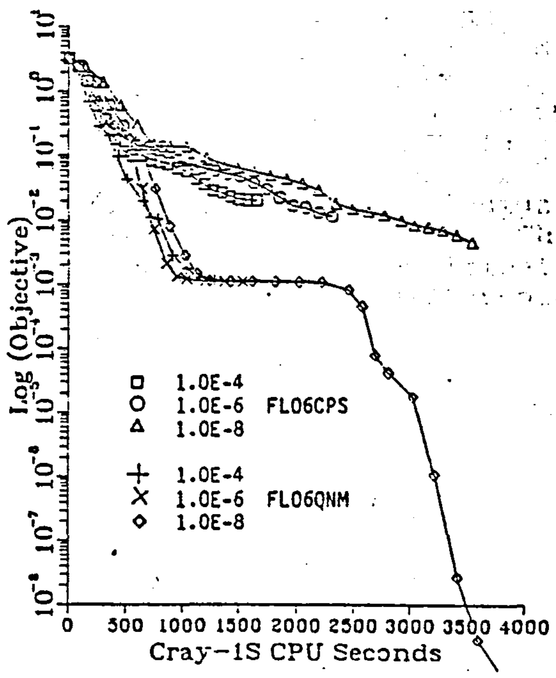

Figure 11. Test Case III iteration history. Design progress vs. computer time for various objective function precisions.

1. Report No. NASA CR- 166497	2. Government Accession No.	3. Recipient's Catalog No.
4. Title and Subtitle IMPROVED METHOD FOR TRANSONIC AIRFOIL DESIGN-BY-OPTIMIZATION		5. Report Date June 1983
		6. Performing Organization Code ECG
7. Author(s) Robert A. Kennelly, Jr.		8. Performing Organization Report No. TN 11555-307-8
9. Performing Organization Name and Address Informatics General Corporation Palo Alto, California		10. Work Unit No. J.O. T3334Y
		11. Contract or Grant No. NAS2-11555
12. Sponsoring Agency Name and Address National Aeronautics and Space Admistration Washington, D.C. 20546		13. Type of Report and Period Covered Contractor Report
		14. Sponsoring Agency Code RTOP # 5053121

15. Supplementary Notes

Point of Contact: Technical Monitor, William P. Jones, MS 233-10
Ames Research Center, Moffet Field CA 94035
(415) 965-6482 or FTS 448-6482

16. Abstract

An improved method for use of optimization techniques in transonic airfoil design is demonstrated. FLO6QNM incorporates a modified quasi-Newton optimization package, and is shown to be more reliable and efficient than the method developed previously at NASA-Ames, which used the COPES/CONMIN optimization program. The design codes are compared on a series of test cases with known solutions, and the effects of problem scaling, proximity of initial point to solution, and objective function precision are studied. In contrast to the older method, well-converged solutions are shown to be attainable in the context of engineering design using computational fluid dynamics tools, a new result. The improvements are due to better performance by the optimization routine and to the use of problem-adaptive finite difference step sizes for gradient evaluation.

17. Key Words (Suggested by Author(s)) Airfoil design, Optimization, Transonic, Computational fluid dynamics, Minimization, Nonlinear programming		18. Distribution Statement Unclassified - Unlimited STAR Category	
19. Security Classif. (of this report) Unclassified	20. Security Classif. (of this page) Unclassified	21. No. of Pages 11	22. Price* --

*For sale by the National Technical Information Service, Springfield, Virginia 22161

End of Document