ARTICLE NO. 79-1529R

J80 - 208 ~~207~~

00001
20005
20018

Technique for Developing Design Tools from the Analysis Methods of Computational Aerodynamics

Warren H. Davis Jr.*
Grumman Aerospace Corp., Bethpage, N.Y.

An initial attempt at a technique to generate, simply and rapidly, mixed direct-inverse codes from their direct counterparts is presented. A surface alteration procedure, keyed to flow type for the mixed case, is "wrapped around" an existing direct code. The difference between calculated and target pressures determines surface modifications. The direct code is used as a "black box." The technique features retention of the strength of the direct analysis and easy interchangeability of the analysis code. The technique is demonstrated in two cases, a nonlinear supersonic wing code and a transonic airfoil code.

Nomenclature

C_p	= pressure coefficient
C_{pi}	= pressure coefficient associated with incompressible flow
e	= error measure for design iteration
ℓ	= surface wavelength for model problem
M	= Mach number
t/c	= airfoil thickness ratio
x	= nondimensional longitudinal coordinate
y	= nondimensional normal coordinate
γ	= ratio of specific heats
Subscripts	
s	= body surface
L	= local
∞	= undisturbed freestream
Superscripts	
$( )', ( )''$	= d/dx, d^2/dx^2
k	= design iteration number
T	= target value
$*$	= critical value

Introduction

THE subject of the work presented here is aerodynamic design codes. Much interest has been generated in the last ten years in computational tools for airfoil and wing design; i.e., the works documented by the 1978 NASA Conference on Advanced Technology Airfoil Research.[1] This is presently the focal area for work with the three-dimensional transonic relaxation codes. But even current transonic airfoil design codes can have difficulties when applied at extreme conditions.

Airfoil design codes generally fall into two categories, the inverse[2] and the mixed direct-inverse.[3-6] The inverse code requires Dirichlet data as the boundary conditions, with the final result being a surface shape. The direct-inverse code solves a mixed boundary value problem. Neumann (direct) data are specified in the region where the shape is to be retained and Dirichlet (inverse) data are specified on the rest of the airfoil where the shape is to be obtained. Unfortunately, such airfoil design methods require large and costly development efforts to obtain a working code. Code updating, to take advantage of advancements in numerical techniques, may again require major efforts and changes to the design code. Most important, however, inverse code performance levels, flow conditions at which a converged solution can be obtained, are somewhat lower than that of the more powerful direct codes.

A technique which systematically adapts the direct codes, as they are, to the design problem would be particularly useful. What is needed is a surface alteration procedure that presents a new surface to the direct code. The resulting code would retain all of the advantages of the direct code while requiring only minimal development time since the major work, the analysis code, is complete and unaltered by the adaption. In this way as refinements to the analysis are made, their incorporation into the design code becomes a simple matter of interchanging analysis codes. This general approach was taken earlier by Barger and Brooks[7] with their streamline curvature method for airfoil design. It will be shown later that their method of surface alteration is basically appropriate to subsonic speeds.

In this paper, this general approach mentioned is taken up again. A somewhat general technique is developed and illustrated by preparing a supersonic three-dimensional wing code and a transonic airfoil code. The surface alteration method is keyed to the local velocity (supersonic or subsonic) for the mixed case and reduces to the Barger and Brooks method for the purely subsonic case. The present method is fully automated into one hands-off computer code while the Barger/Brooks method is implemented through a series of computer runs sharing a common analysis program.

Technique

The basic procedure to generate the design code is diagrammed in Fig. 1. The starting point is an existing analysis code. The design iterations are then built around this code as geometry alterations followed by analysis runs. The computing time for a complete design run can be held reasonably low by not requiring convergence for any given analysis. Convergence is now obtained through the number of design iterations. The resulting code is in the form of an iterative-direct technique, presumably retaining all the strengths of the original analysis.

The geometry alteration package requires some rule by which the surfaces are changed. The driver for these modifications is the difference in pressure coefficients between the calculated and target pressures. Since convergence will be attained through a series of relaxed iterations, this rule need not be extremely accurate; in fact, it may even be quite

Presented as Paper 79-1529 at the AIAA 12th Fluid and Plasma Dynamics Conference, Williamsburg, Va., July 23-25, 1979; submitted Aug. 23, 1979; revision received March 13, 1980.

Index categories: Transonic Flow; Computational Methods; Aerodynamics.

*Engineer, Aerodynamics Section. Member AIAA.

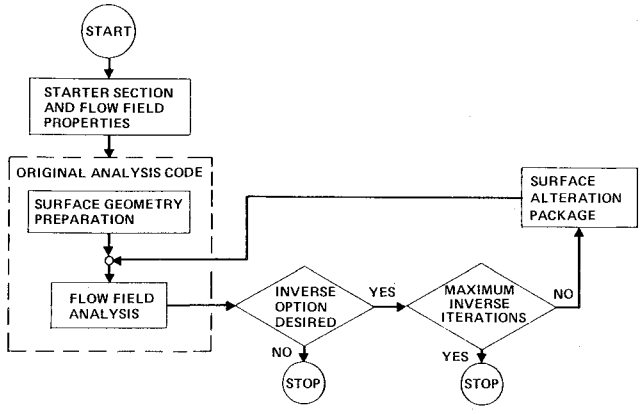

Fig. 1 Automated iterative-direct design technique.

crude provided it is sound. In addition, since the search is for a relation between variations, the expression need only be of the perturbation type. Guidance as to the type of rule expected may be obtained from the wavy-wall problem,[8] which in small disturbance theory gives:

Supersonic

$$C_p = (2/\sqrt{M_\infty^2 - 1})\, y_s' \tag{1}$$

Subsonic

$$C_p = \ell/\pi\,(1/\sqrt{1-M_\infty^2})\, y_s'' \tag{2}$$

Although these are gross simplifications, they provide a broad idea of a method upon which an under-relaxed iterative scheme may be based.[9] The idea is to interpret M_∞ as the local Mach number M_L. In a mixed flowfield, surface slope changes in the supersonic region can be matched to required pressure changes through Eq. (1). The subsonic portion of the field requires alteration of the surface second derivatives.

It is noted, at this point, that the iterative-direct method of Barger and Brooks[7] uses a single equation that relates the pressures to surface curvatures. The wavy-wall model suggests this method to be of a subsonic nature. The fact that their method works quite well for moderate supercritical sections strengthens the position that a rather crude rule for surface alterations may be all that is required. The suggestion of two equations for the mixed problem is also an interesting analogy to the separate differencing schemes required in the handling of transonic relaxation numerics.[10]

Equations (1) and (2) are cited simply as guidance and certainly are quantitatively crude. However, when the local Mach numbers are substantially away from unity, one is tempted to try them as they are.[9] One such case where this method has been carried out successfully for supersonic flow is described below. For this purely supersonic case, the surface alteration scheme is carried out using Eq. (1) as follows:

$$(y')^k = (y')^{k-1} + \left(\frac{dy'}{dC_p}\right)_{\text{Eq. (1)}} (C_p^T - C_p^{k-1}) \tag{3}$$

For M_L close to 1.0 the previous equations are unusable. A wavy-wall model reflecting transonic effects would be required as a counterpart of Eqs. (1) and (2). In line with the fact that the rules need not be exact for the method, the work of Spreiter and Alksne[11] is followed. These are local linearizations of the transonic equation to obtain independently subsonic and supersonic thin airfoil solutions. The resulting equations are:

$M_\infty > 1$:

$$C_p = \frac{2}{M_\infty^2(\gamma+1)} \times \left\{(M_\infty^2-1) - \left[(M_\infty^2-1)^{3/2} - \frac{3}{2}M_\infty^2(\gamma+1)y_s'\right]^{2/3}\right\} \tag{4}$$

$M_\infty < 1$:

$$C_p = \frac{-2}{M_\infty^2(\gamma+1)} \times \left\{(1-M_\infty^2) - \left[(1-M_\infty^2)^{3/2} + \frac{3}{4}M_\infty^2(\gamma+1)C_{p_i}\right]^{2/3}\right\} \tag{5}$$

Equation (4) explicitly relates local pressures to the surface slopes as in Eq. (1). Its counterpart, Eq. (5), is not in a comparable form in that it contains the incompressible airfoil integral

$$C_{p_i} = -\frac{2}{\pi}\int_0^c \frac{dy/d\xi}{x-\xi}\,d\xi \tag{6}$$

and no explicit appearance of y_s'' as in Eq. (2). Since the main interest is to under-relax a relatively crude surface alteration rule, the incompressible integral is replaced by the subsonic wavy-wall solution, Eq. (2), with $M_\infty = 0$ and $\ell = c$ (= chord). Thus,

$$C_{p_i} \to (c/\pi)\, y_s'' \tag{7}$$

Equations (4), (5), and (7), usable for transonic flow, explicitly relate surface first and second derivatives to local pressure coefficients.

The problem is to interpret M_∞. The explicit local nature of Eqs. (1) and (4) suggest that in regions where $M_L > 1$ the local Mach number is correct. Equations (2) and (5) require more thought. Both the integral form of C_{p_i} and its crude approximation are global in character. The integral represents the local C_p as a weighted integral over the full chord, while its wavy-wall approximation retains the global length parameter (replaced here by the chord).

The choice of reference Mach number for Eq. (5) is further complicated since Spreiter's equations are keyed to switch character (subsonic/supersonic) on the critical pressure coefficient. In this sense, the small disturbance character of Eqs. (4) and (5) may not be compatible with the flowfield calculations performed by the analysis code. For the transonic case to be discussed below, the full potential equation is used rather than some small disturbance approximation. It seems more reasonable to redefine this character switch on C_p^* to be compatible with the full potential representation. This can be accomplished by redefining M_∞ to be used when $M_L < 1$ as M_∞^* such that C_p^* is calculated consistently with the full-potential equation.

$$C_p^* = \frac{2}{\gamma M_\infty^2}\left\{\left[\left(\frac{\gamma+1}{2}\right)\Big/\left(1+\frac{\gamma-1}{2}M_\infty^2\right)\right]^{\gamma/1-\gamma} - 1\right\} \tag{8}$$

The switching is then correct for

$$M_\infty^* = 1\Big/\left[1-\left(\frac{\gamma+1}{2}\right)C_p^*\right] \tag{9}$$

The final form of Eqs. (4) and (5) to be used for the transonic surface alterations is

$M_L > 1$:

$$C_p = \frac{2}{M_L^2(\gamma+1)} \times \left\{ (M_L^2 - 1) - \left[(M_L^2-1)^{3/2} - \frac{3}{2} M_L^2 (\gamma+1) y_s' \right]^{2/3} \right\} \quad (10)$$

$M_L < 1$:

$$C_p = \frac{-2}{M_\infty^{*2}(\gamma+1)} \times \left\{ (1-M_\infty^{*2}) - \left[(1-M_\infty^{*2})^{3/2} + \frac{3}{4} M_\infty^{*2} (\gamma+1) \frac{c}{\pi} y_s'' \right]^{2/3} \right\} \quad (11)$$

$M_L = 1$ not calculated

These equations are used to calculate gradients by which the ΔC_p's generate first and second derivative changes to the surface. The surface alterations at each iteration are carried out as

$M_L < 1$:

$$(y'')^k = (y'')^{k-1} + \left(\frac{dy''}{dC_p} \right)_{\text{Eq. (11)}} (C_p^T - C_p^{k-1}) \quad (12)$$

$M_L > 1$:

$$(y')^k = (y')^{k-1} + \left(\frac{dy'}{dC_p} \right)_{\text{Eq. (10)}} (C_p^T - C_p^{k-1}) \quad (13)$$

Integration yields the new surface. For the mixed flow case these integrations are carried out as shown in Figs. 2 and 3. Boundary conditions are applied to maintain continuity at the initial point of modifications and the shock point.

The geometry alteration package requires the construction of arrays of surface first and second derivatives from airfoil ordinates. These derivatives are obtained by sequencing three point geometric constructions for five conic sections about each point. The five resulting derivatives at the central point are then averaged to yield a final derivative for the central point. This algorithm is repeated on the set of slopes to obtain surface second derivatives.

After redefinition of the surface derivatives using Eqs. (12) and (13), an integration scheme is required to recover the airfoil ordinates. The algorithm employed is a running Simpson's rule for each interval. Each interval is subdivided into eleven subintervals using a two point-two slope cubic geometric construction overlaying four intervals with the division occurring in the central interval. Exterior intervals are subdivided using a parabolic construction with a three point overlay.

Supersonic Example: Nonlinear Code for Supersonic Wings

The first application of the technique involved the generation of a nonlinear design code for supersonic wings from the corresponding direct code. The intent is to prescribe the geometry on the lower surface and in the leading edge region. The pressure is prescribed on the rest of the upper surface where the geometry is to be obtained.

The direct code to be used is the pilot code called COREL developed at Grumman by Grossman.[12] This nonlinear code handles conical supersonic wings with camber and incidence. The streamwise flow is supersonic with the formation of a bow shock, while the crossflow may be transonic with embedded shock waves (see Fig. 4). Since the problem is supersonic in the streamline direction, only the supersonic wavy-wall formula, Eq. (1), is required if surface slope changes are made following those streamlines.

The mechanics of the surface slope changes follow. A target pressure distribution is chosen as input to drive the surface modifications. A converged pressure distribution is then

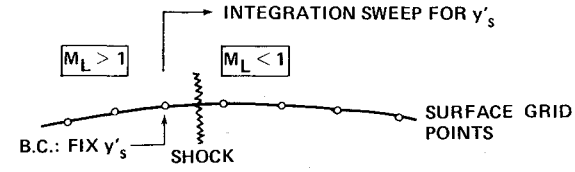

Fig. 2 Mechanics of mixed surface modifications; junction of regions ($M_L > 1, M_L < 1$).

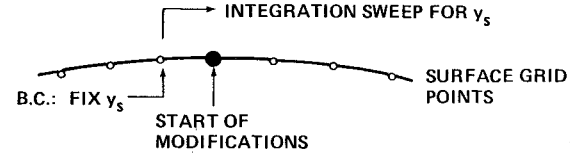

Fig. 3 Mechanics of mixed surface modifications; beginning of modified region.

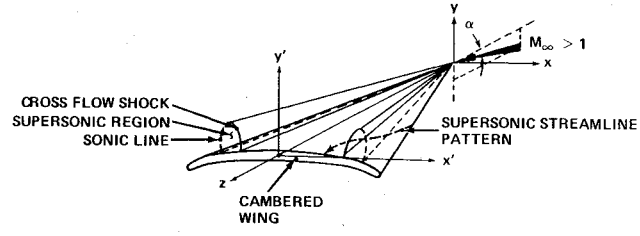

Fig. 4 Cambered wing in supersonic conical flow showing the transonic crossflow plane (x', y').

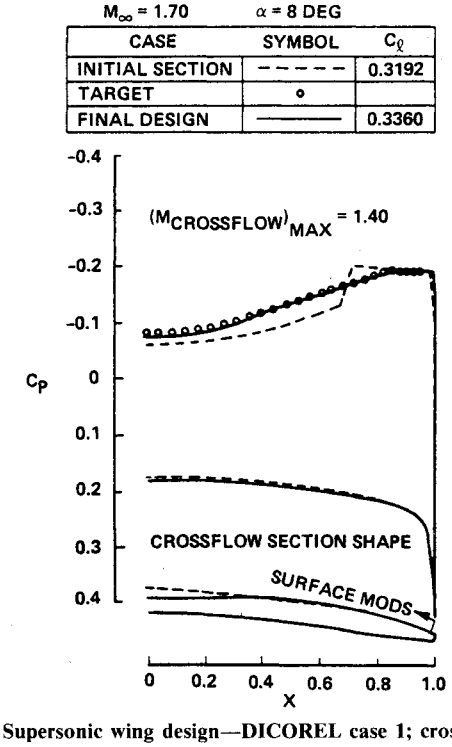

Fig. 5 Supersonic wing design—DICOREL case 1; crossflow plane pressure distributions.

calculated over the initial surface. Referring to Fig. 4, a crossflow sweep of the upper surface is made from a user chosen point, say the leading edge, inboard to the wing plane of symmetry (y', Z plane). At each grid point, Eqs. (1) and (3) are combined and applied to alter the surface locally in the streamline direction. The new surface slopes are then transferred to the crossflow plane (x', y' plane) and the new wing is reconstructed by a single integration of these slopes. The resulting code, DICOREL (direct-inverse COREL), represents only a minor modification of COREL; its main structure remains intact.

The resulting application of DICOREL to a cambered wing is shown in Fig. 5. The initial crossflow section produced a shock on the upper surface which was strong enough to possibly produce boundary-layer separation. The target pressure distribution was chosen for two reasons: first, to produce a shockless crossflow to delay separation; second, to lower the pressure over the inboard region. The first point to be modified is located at station $(1-x)=0.03$. As Fig. 5 shows, a major changes to the upper surface have been calculated. Agreement between target and final analysis pressures for the new section is excellent.

An error measure is defined as follows

$$e=\frac{1}{N}\sum_i |C_{p_i}^T - C_{p_i}| \quad N=\text{number of points}$$

and is plotted in Fig. 6. For this case, it shows a monotonic decay as the iterations continue. Note that the major changes occur rapidly; see the first 10-15 iterations. The average pressure error at a point is 1%.

A typical computation time for a COREL analysis with 60 (on the section) $\times$ 30 grid points is 3 min on an IBM/370. DICOREL requires approximately 6 min to perform 20 design iterations. This includes the time required to attain convergence for the analysis of the initial section. The total running time is kept low by performing only 30 relaxation cycles of each analysis after each surface alteration. The convergence for the final section is obtained cumulatively through the number of design iterations.

The second DICOREL example, Fig. 7, shows upper surface modifications starting at the leading edge. The first upper surface point to be modified is located at $(1-x)=0.001$. The target pressure distribution is chosen to first remove the leading edge suction peak, and second to weaken the crossflow shock. Both alterations were chosen to lessen the occurrence of separation in their respective regions. The design was carried out on a fine gid (100 points on the section surface) to gain resolution for both areas. Agreement of target and analysis pressures at both leading edge and shock position is very good. In both regions, the approach to the target is made monotonically. The actual surface changes are too small to be seen in Fig. 7. The local differences in crossflow surface slopes between the original and the final sections are plotted in Fig. 8. Note again that the changes are local in character and begin directly at the leading edge.

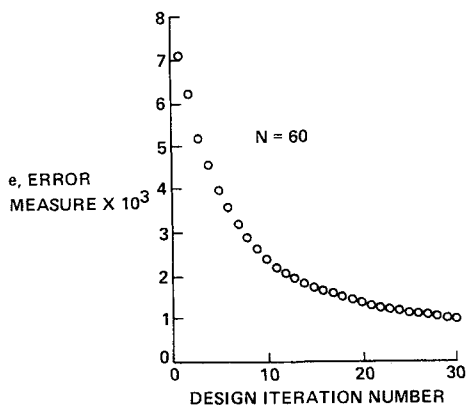

Fig. 6 Convergence history for supersonic wing design, case 1 (60 $\times$ 60 grid).

Transonic Example: Airfoil Code

A second application of the technique involves the generation of a transonic airfoil design code. The goal is to develop a method that accepts a target pressure distribution on the upper surface starting at a desired station near the leading edge. Surface alterations are made from this point aft to the trailing edge. The remaining airfoil ordinates are fixed. The direct code to be exploited is the powerful circle-plane relaxation code for the full potential equation by Jameson.[13] This problem tests the technique in a mixed subsonic-supersonic flowfield.

Since surface alterations are done transonically, the wavy-wall model reflecting transonic effects is appropriate. Equations (10) and (11) are used. The supersonic solution relates the ΔC_p's to a surface slope change, the subsonic, to a surface second derivative change. These rules are applied in their respective regions to obtain surface modifications using Eqs. (12) and (13).

The first two examples to be shown deal with the ability of the present code to accurately obtain a known section. When pressure results from a direct calculation over a known airfoil are used as target values for the design calculations, the original airfoil shape should be recovered. For these cases, a

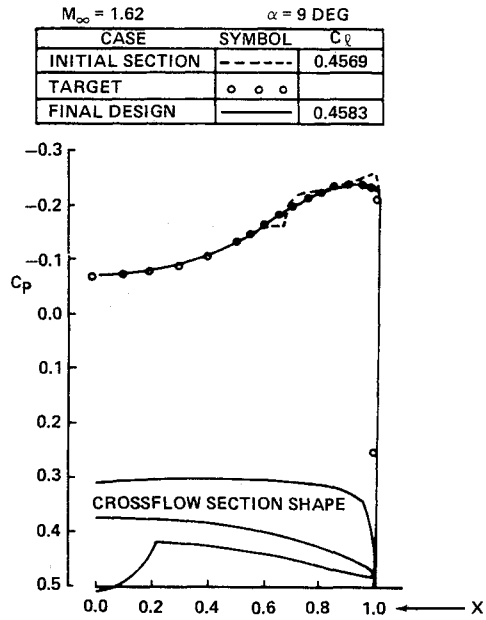

Fig. 7 Supersonic wing design—DICOREL case 2; crossflow plane pressure distributions.

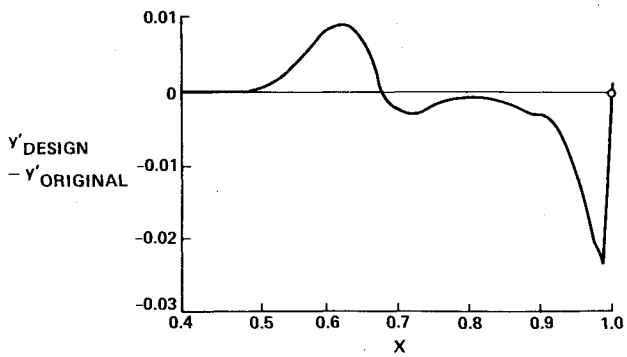

Fig. 8 DICOREL case 2: upper surface crossflow slope changes, supersonic wing design.

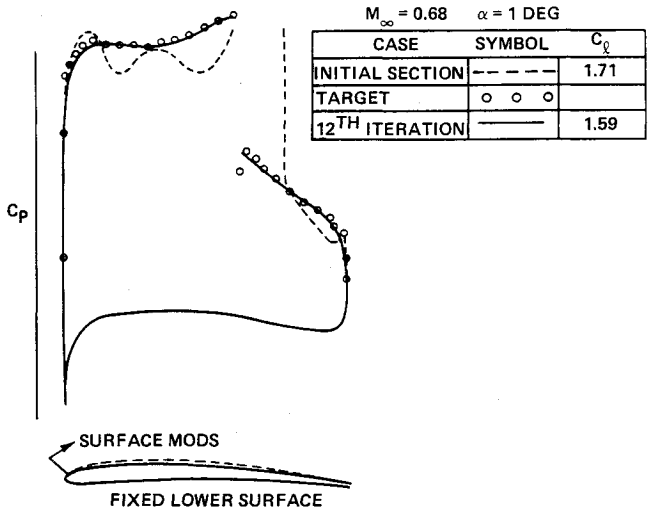

Fig. 9 Transonic section design, test case 1.

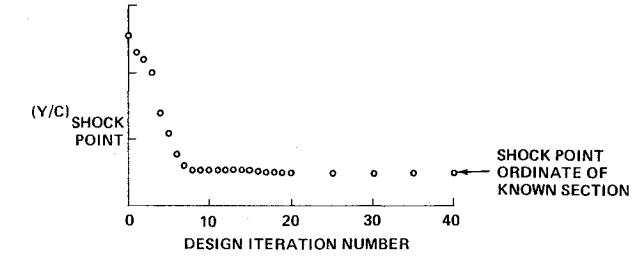

Fig. 10 Transonic design test case 1, convergence history of shock point ordinate.

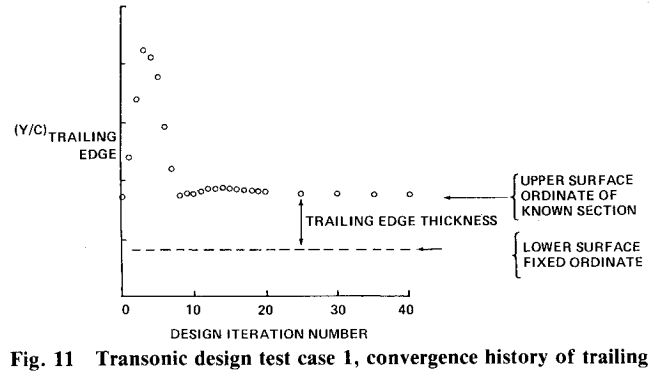

Fig. 11 Transonic design test case 1, convergence history of trailing edge ordinate.

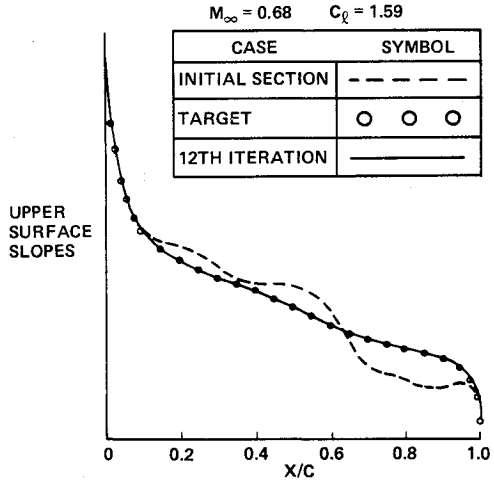

Fig. 12 Transonic design test case 1, comparison of surface slopes.

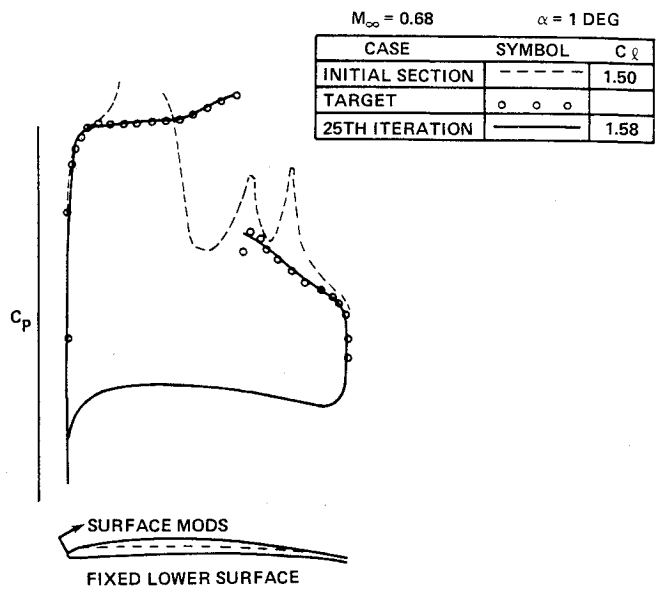

Fig. 13 Transonic section design, test case 2.

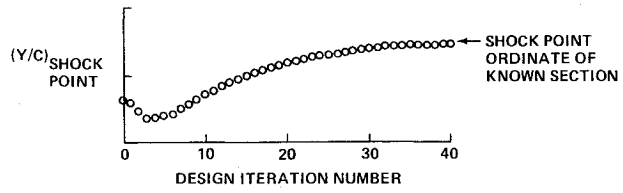

Fig. 14 Transonic design test case 2, convergence history of shock point ordinate.

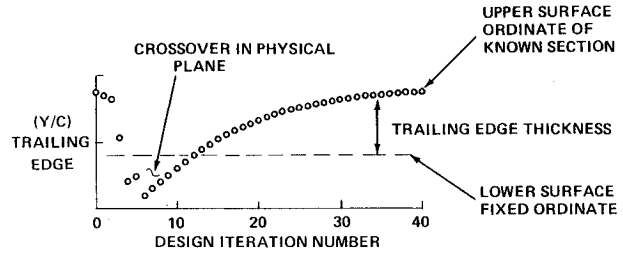

Fig. 15 Transonic design test case 2, convergence history of trailing edge ordinate.

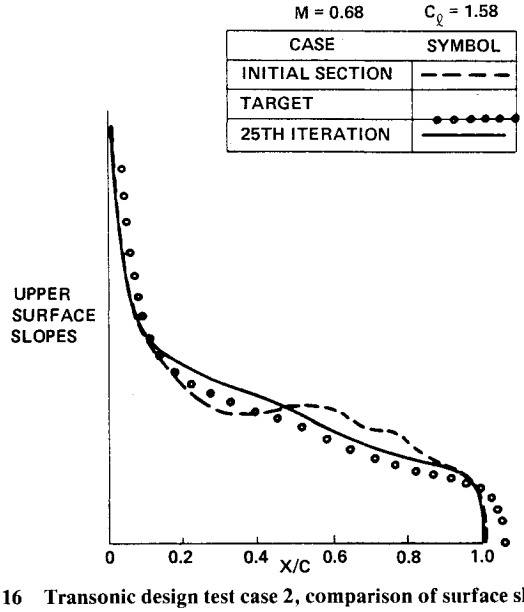

Fig. 16 Transonic design test case 2, comparison of surface slopes.

high performance transonic section is chosen as the target airfoil with the maximum Mach number before the shock greater than 1.4 when $M_\infty = 0.68$. The section has a maximum thickness ratio of approximately 6%. The direct calculation of the pressures is performed at a final grid size with 96 points on the airfoil surface. This now becomes the target pressure distribution.

In Fig. 9, the first case, the initial section to begin the design process is much thicker than the target section and displays a very strong shock at the 80% chord station. The surface modifications begin at the 5% chord station. After 12 design iterations the target pressures are achieved.

A more stringent test than comparison of pressure distributions is the comparison of surface ordinates or even slopes. Figure 10 shows the movement with each design iteration of the shock point ordinate. This ordinate is brought to the target value within 10 iterations. Note the stable behavior for the next 30 iterations. Similarly, Fig. 11 shows the same graph for the upper surface trailing edge ordinate. The target value is again reached within 10 design iterations. The upper surface slopes are shown in Fig. 12 for the 10th design iteration. Correlation is again very good.

A typical computation time for the Jameson analysis code is 2.5 min on an IBM/370. In the iterative-direct mode a design run, including convergence on the initial section, requires approximately 4 min for 20 design iterations.

The second test case uses the same target section and pressure distribution but starts with a much thinner airfoil. More importantly, the surface modifications start very close to the leading edge, at the 0.6% chord station. The resulting pressure distributions of Fig. 13 show that even though the initial section had three shocks on the upper surface, the target pressures are again attained through the design process. In Figs. 14 and 15, the movement of both shock point and trailing edge ordinate show agreeable behavior and very good comparison to the known values. Finally, in Fig. 16, the surface slopes for design iteration number 25 show excellent agreement with the target airfoil slopes in both sensitive regions, the leading edge and the shock point.

The two previous cases demonstrate that the technique yields a stable design code even when applied to high performance transonic sections. The next case deals with modifying a section to obtain an arbitrary pressure distribution. Figure 17 shows the previous transonic section, now being used as the section to be modified, operating at a C_l of 1.4. The target pressure distribution is chosen to weaken the leading edge expansion, carry the load slightly more forward, and weaken the shock. Surface modifications begin at the 0.6% chord station. Twenty-five design iterations produced the final modified section and pressure distribution in Fig. 17. Minimal changes are required of the section shape; maximum thickness remains essentially unchanged while the trailing edge is opened by approximately 15%.

One of the initial rationales for developing this technique was the ability to exploit the strengths of the analysis codes to obtain solutions at performance levels which could not be attained with the present inverse codes. One such mixed direct-inverse method is the transonic section design code TRANDES,[3,14] developed by Carlson. TRANDES generally produces useful results but, as with most codes, can be pushed to the point where no converged solution can be obtained even where the solution is known to exist. The final example to be discussed compares TRANDES to the present method for just such a case. The attempt is to use each design code to recover a known section when the target pressures were calculated by the corresponding analysis code. Initial attempts with the TRANDES code revealed that for equivalent levels of accuracy the inverse required a finer grid than the direct. This is not an unusual situation. Thus, to be optimistic in the approach, the finest grid available was chosen for the test. The resulting grid for each code placed 97 points on the section surface. This approach was also preferred since it produced an accurate description of the pressure distribution in the important areas, the leading edge, trailing edge, and shock region. The conditions for the test case were $C_\ell = 1.4$ at $M_\infty = 0.683$, $(t/c)_{\max} \approx 6\%$.

The results of the TRANDES test are shown in Fig. 18. The important point to note is that although a converged direct solution is obtained on the fine grid, the resulting pressure distribution used as input to the inverse form of the code would not yield a solution on that same grid. The strengths of the analysis code do not seem to be retained by its design counterpart at this level of airfoil performance.

Since the fine grid TRANDES inverse run was incomplete, the intermediate results are also shown in Fig. 18. The pressure results are shown for the 49×25 grid. While reasonably good pressure distribution comparison is seen, the modified section shows discrepancies with the target section for both upper and lower surfaces. The attempt to continue the calculation into the final grid (97×49) diverges, with the surfaces crossing at the trailing edge before the calculation stops prematurely. This type of response is repeated or worsens at high C_ℓ's.[15]

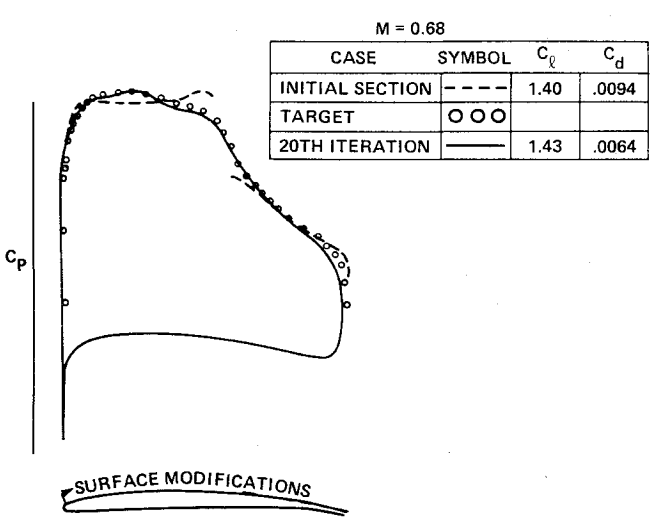

Fig. 17 Transonic section design.

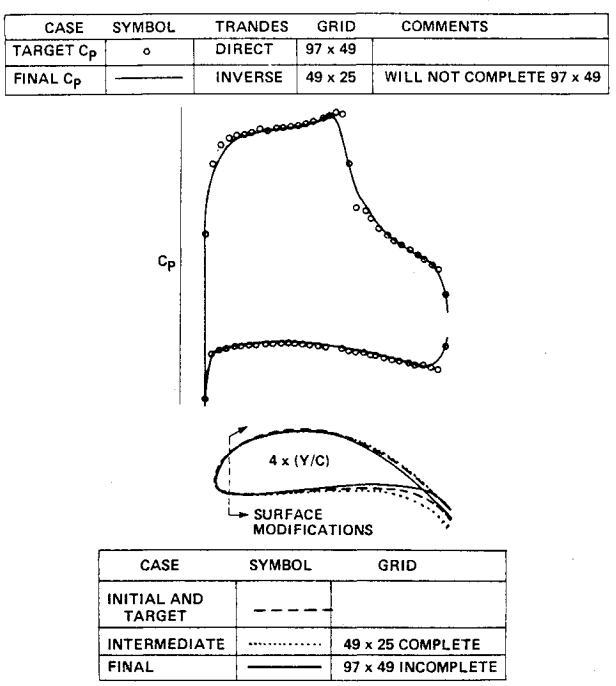

Fig. 18 TRANDES, attempt to recover initial section at $C_\ell = 1.4$, $M_\infty = 0.683$.

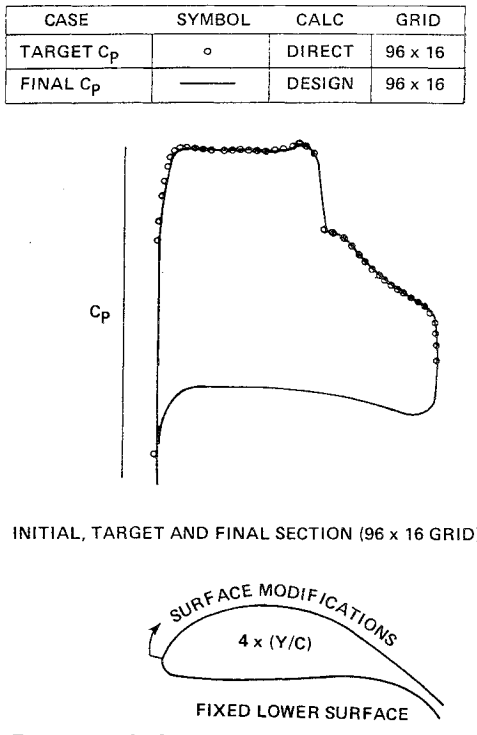

Fig. 19 Present method, attempt to recover initial section at $C_l = 1.4$, $M_\infty = 0.683$.

Figure 19 shows the corresponding results obtained using the present method. The target pressures are attained in the 96×16 grid by the design process. The differences in the initial and final section shapes cannot be discerned on the 4-times ordinate magnification shown in the figure. In addition, designs have been obtained with the present method for similar transonic sections at C_l's as high as 2.0 for an even finer grid (192×32). The strengths of the analysis code appear to be retained in the present design code, at least to this level of performance.

It must be stated that this comparison of methods is not as straightforward as appears. The TRANDES calculation fixes the nose region up to the 7% chord station and allows alterations to both upper and lower surfaces aft of that point. The calculation with the present method begins alterations at the 0.3% chord station but the entire lower surface is fixed. So each method contains a different type of geometric constraint. The effect of these different constraints is not completely known, but both would seem to allow a large degree of flexibility; TRANDES with the two surfaces and the present method with the very sensitive nose region. Figures 18 and 19 should be compared with this point in mind. It is worth noting that the application of the present technique to the lower surface, or a combination of both upper and lower is presently being investigated.

Discussion

This work should be considered as an initial attempt at the present conversion technique. The main concern has been to obtain a tool to perform modifications to surfaces which retain as much of the strength of the original analysis code as possible. Questions which generally arise in these inverse problems, such as whether or not the problem is well posed or the correct treatment of the shock point have not been specifically addressed up to this point. In spite of this, the two applications of the technique have been quite successful.

As with any inverse or direct-inverse method, there remain, for this technique, several questions regarding the constraints on the target pressure distribution. For example, the simultaneous prescription of an arbitrary pressure distribution and a freestream Mach number may be inconsistent. This problem is not solved by the present technique, but its effects may have been softened by the iteration procedure chosen. The iterative-direct method uses "elemental" iterations in which the prescribed shape and freestream Mach number are totally consistent within an iteration. A solution will be obtained. The inconsistency is now placed between each geometry iteration. So far this has not placed any restrictions on the usefulness of the codes.

Any code which designs to a specified pressure distribution can run into other computing difficulties. In particular, problems can arise with trailing edge thickness. In this respect, the ability of the present technique to exploit a powerful direct code may have an additional benefit. Jameson's direct code used for the transonic case does not appear to have any difficulty analyzing sections with either large positive or negative trailing edge thicknesses. This is because the mapping to the circle plane implicitly contains the required source term such that there is no overlapping of regions in the computational plane.

The starting point for the surface alterations of the present technique separates regions of direct and inverse calculations, the result is a mixed boundary value problem. This is true even though the main calculation is done in a direct fashion. The problems associated with this point, as discussed by Woods,[16] have not been treated here. This position is taken simply because problems have not been encountered with the formulation as presented (see Fig. 3). Similarly at a shock point, no attempt has been made to deal with the singularity to be expected analytically at the foot of the shock. No other provisions at either point are made other than to insure continuity of surface (see Fig. 2). Since no special provisions are required by the analysis code, this was the initial position taken with respect to the design code. The results have been quite satisfactory as previously shown.

It is interesting to note that Jameson has recently updated[17] the FLO 6 code in fully conservative form to incorporate a multiple grid method to accelerate the calculations. The resulting computing times have been quoted as being an order of magnitude smaller than the original FLO 6 calculation. Since the old FLO 6 code is used in the transonic code presented here, this provides the ideal opportunity to exploit the "black box" feature of the technique. This new analysis code may be simply substituted into the transonic airfoil code. The expectation is to attain computation times for the full section design process of under 1 min.

Conclusions

The general technique described above leads to a systematic and simple generation of iterative-direct design codes from their direct counterparts. The results presented are encouraging as initial attempts at the technique in that: 1) minimal time and effort are required for the conversion; 2) analysis codes are easily interchangeable; 3) technique has been successfully applied to two codes; 4) strengths of the analysis code appear to be retained in the design version; and 5) computation time can be kept reasonably low (approximately double the analysis counterpart).

Acknowledgment

This work was conducted under Grumman Advanced Development Program funding. The author gratefully acknowledges the technical support and guidance of his colleagues at Grumman, in particular, G. DaForno, W. Mason, and A. Vachris Jr.

References

[1] *Advanced Technology Airfoil Research*, Proceedings ATAR Conference, NASA CP 2045, March 1978.

[2] Volpe, G. and Grossman, B., "The Analysis and Design of Transonic Two-Element Airfoil Systems," *Proceedings of the Conference on Advanced Technology Airfoil Research*, NASA CP 2045, Vol. 1, 1978, pp. 209-220.

[3] Carlson, L. A. and Richoll, B. M., "Application of Direct-Inverse Techniques to Airfoil Analysis and Design," *Proceedings of the Conference on Advanced Technology Airfoil Research*, NASA CP 2045, Vol. 1, 1978, pp. 55-72.

[4] Shankar, V., Malmuth, N. D., and Cole, J. D., "A Consistent Design Procedure for Supercritical Airfoils in Free Air and a Wind Tunnel," *Proceedings of the Conference on Advanced Technology Airfoil Research*, NASA CP 2045, Vol. 1, 1978, pp. 101-117.

[5] Eggleston, B., "An Inverse Method for the Design of Airfoils with Supercritical Flow," SAE Paper 770450, Wichita, Kan., March 1977.

[6] Tranen, T. L., "A Rapid Computer Aided Transonic Airfoil Design Method," AIAA Paper 74-501, Palo Alto, Calif., June 1974.

[7] Barger, R. L. and Brooks, C. W., "A Streamline Curvature Method for Design of Supercritical and Subcritical Airfoils," NASA TN D-7770, Sept. 1974.

[8] Liepmann, H. W. and Roshko, A., *Elements of Gas Dynamics*, John Wiley and Sons, New York, 1957.

[9] Mason, W., private communication.

[10] Murman, E. M. and Cole, J. D., "Calculation of Plane Steady Transonic Flows," *AIAA Journal*, Vol. 9, 1971, pp. 114-120.

[11] Spreiter, J. R. and Alksne, A. Y., "Thin Airfoil Theory Based on Approximate Solution of the Transonic Flow Equation," NACA Rept. 1359, 1958.

[12] Grossman, B., "A Numerical Procedure for the Computation of Supersonic Conical Flows," AIAA Paper 78-1213, Seattle, Wash., July 1978.

[13] Jameson, A., "Iterative Solution of Transonic Flows Over Airfoils and Wings, Including Flows at Mach 1," *Communications on Pure and Applied Mathematics*, Vol. 27, 1974, pp. 283-290.

[14] Carlson, L. A., "TRANDES: A Fortran Program for Transonic Airfoil Analysis or Design," NASA CR-2821, 1977.

[15] DaForno, G. and Kohn, J., private communication.

[16] Woods, L. C., "The Design of Two-Dimensional Airfoils with Mixed Boundary Conditions," *Quarterly of Applied Mathematics*, Vol. 13, 1955, pp. 139-149.

[17] Jameson, A., "Acceleration of Transonic Potential Flow Calculations on Arbitrary Meshes by the Multiple Grid Method," AIAA Paper 79-1458, Williamsburg, Va., July 1979.