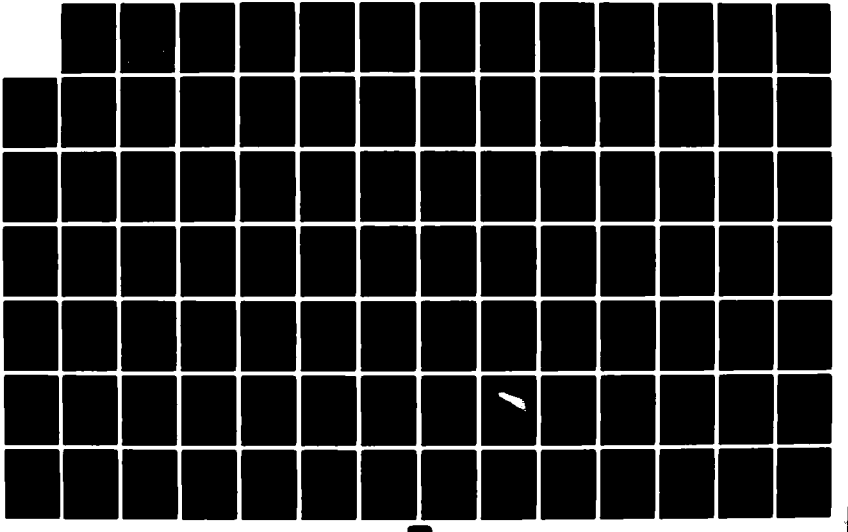
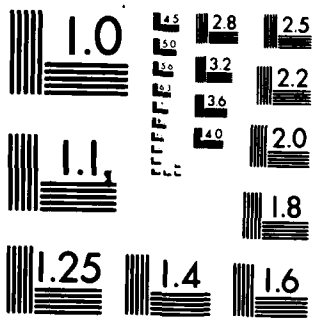
AD-A129 573 COMBINED DIRECT/INVERSE THREE-DIMENSIONAL TRANSONIC WING DESIGN(U) LOCKHEED-GEORGIA CO MARIETTA R A WEED ET AL. MAY 83 LG83-ER-0060 1/2

UNCLASSIFIED DTNSRDC/ASED-CR-03-83 N00167-81-C-0078 F/G 20/4 NL





DTNSRDC-ASED-CR-03-83

COMBINED DIRECT/INVERSE THREE-DIMENSIONAL TRANSONIC WING DESIGN

Richard A. Weed
Lockheed-Georgia Company
Marietta, Georgia 30063

Leland A. Carlson and William K. Anderson
Texas A&M University
College Station, Texas 77843

May 1983

Final report for period May 1981 to September 1982

APPROVED FOR PUBLIC RELEASE: DISTRIBUTION UNLIMITED

Prepared for

DAVID W. TAYLOR NAVAL SHIP RESEARCH AND DEVELOPMENT CENTER
Aviation and Surface Effects Department
Bethesda, Maryland 20084

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE

READ INSTRUCTIONS BEFORE COMPLETING FORM

1. REPORT NUMBER: DTNSRDC-ASED-CR-03-83	2. GOVT ACCESSION NO.: AD-A129573
3. RECIPIENT'S CATALOG NUMBER:	
4. TITLE (and Subtitle): COMBINED DIRECT/INVERSE THREE-DIMENSIONAL TRANSONIC WING DESIGN	5. TYPE OF REPORT & PERIOD COVERED: Final Report May 1981 - September 1982
	6. PERFORMING ORG. REPORT NUMBER: LG83-ER-0060
7. AUTHOR(s): Richard A. Weed, Leland A. Carlson,* and William K. Anderson*	8. CONTRACT OR GRANT NUMBER(s): N00167-81-C-0078-P00001
9. PERFORMING ORGANIZATION NAME AND ADDRESS: Lockheed-Georgia Company, 86 South Cobb Drive, Marietta, GA 30063	10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS:
11. CONTROLLING OFFICE NAME AND ADDRESS: David Taylor Naval Ship R&D Center, Aviation and Surface Effects Department, Bethesda, MD 20084	12. REPORT DATE: May 1983
	13. NUMBER OF PAGES: 104
14. MONITORING AGENCY NAME & ADDRESS(if different from Controlling Office):	15. SECURITY CLASS. (of this report): UNCLASSIFIED
	15a. DECLASSIFICATION/DOWNGRADING SCHEDULE:

16. DISTRIBUTION STATEMENT (of this Report)

APPROVED FOR PUBLIC RELEASE: DISTRIBUTION UNLIMITED

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

* Texas A&M University
College Station, TX 77843

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Transonic Flow Analysis
Inverse Wing Design
Computational Aerodynamics
Finite Differences

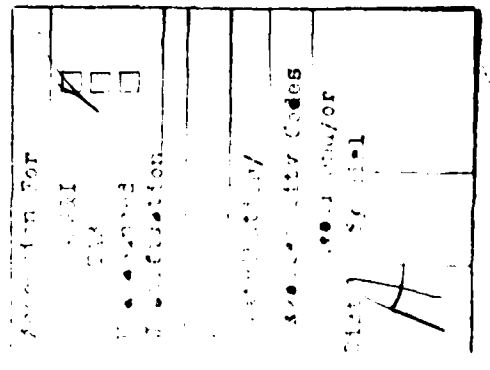
20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

A combined direct/inverse three-dimensional transonic wing design method is presented. The method is built around the ZEBRA II transonic potential flow solution algorithm to provide a design method that is particularly suited for use on a vector computer. The development of a pilot design computer code and a baseline design/analysis code is described. Results are presented that verify the accuracy and consistency of the design method.

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE

UNCLASSIFIED
SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

TABLE OF CONTENTS



LIST OF FIGURES

NOMENCLATURE

a	Speed of sound
A,B,C	Coefficients used in potential extrapolation
$\bar{A},\bar{B},\bar{C}$	Coefficients used in inverse boundary condition
c	Chord
C_D	Drag coefficient
C_L	Lift coefficient
C_m	Moment coefficient
C_1	Constant in baseline inverse boundary condition
G	Perturbation potential
i,j,k	Grid indicies
J	Jacobian
M	Mach number
P	Pressure
q	Velocity magnitude
R	Residual
u,v,w	Cartesian velocity components
U,V,W	Contravariant velocity components
U_∞	Freestream velocity
x,y,z	Cartesian coordinates
X,Y,Z	Computational coordinates
α	Angle of attack
β	Temporal damping coefficients
γ	Gas constant
Γ	Circulation
μ,ν	Retarded density switch functions
η	Spanwise distance from root divided by semispan
ρ	Density
ϕ	Velocity potential
ω	Acceleration parameter

ABSTRACT

A combined direct/inverse three-dimensional transonic wing design method is presented. The method is built around the ZEBRA II transonic potential flow solution algorithm to provide a design method that is particularly suited for use on a vector computer. The development of a pilot design computer code and a baseline design/analysis code is described. Results are presented that verify the accuracy and consistency of the design method.

ADMINISTRATIVE INFORMATION

The work presented was a joint effort by Lockheed-Georgia Company and Texas A&M University supported by the Naval Air Systems Command under the cognizance of D. G. Kirkpatrick (NAVAIR-311D), Navy Contract N00167-81-C-0078-P00001. The authors acknowledge Dr. Tsze C. Tai, contract monitor at the David Taylor Naval Ship Research and Development Center, and Mr. Jerry South, NASA-Langley Research Center, for providing computer time.

INTRODUCTION

In recent years, the increasing importance of transonic flight by both military and commercial aircraft has prompted a large amount of research to develop more accurate and reliable computational methods for the analysis of aircraft configurations in transonic flow. This research was spurred by the increasing costs of wind tunnel tests and the interference and scale problems associated with tests conducted at transonic conditions. As a result of this effort, several computer codes have been developed to calculate the transonic flow about wing and wing-body configurations. A few of these codes have demonstrated levels of accuracy and reliability that have gained them acceptance in the aircraft industry as useful analysis tools.

Unfortunately, the development of wing design codes has lagged the development of analysis codes. The research undertaken in the past few years to develop more efficient wing design methods has centered on two different approaches to the design problem--numerical optimization and

inverse design. Numerical optimization provides a means of automating the trial-and-correction design process using analysis methods. In theory, optimization allows the designer to specify a quantity to be minimized, such as drag, without prior knowledge of the flow details that will produce the objective design. However, most (if not all) transonic codes cannot predict drag accurately enough to use it as a design objective in the optimization process. Therefore, the difference between a specified design pressure distribution and the computed pressure distribution at a span station is used as the function to be minimized in the optimization process. However, the amount of computer time required for optimization limits the technique to performing the design at one span station at a time. In addition, a considerable amount of computer expertise is required to effectively implement the optimization procedure.

In the inverse approach, the wing geometry is computed by specifying a desired pressure distribution over a part of the wing and then solving a mixed Neumann and Dirichlet boundary value problem by finite difference techniques. Since more than one span station at a time can be designed by the inverse technique, it would appear to be much simpler to use and cost less than the numerical optimization procedure.

The present combined direct/inverse transonic wing design program is a joint effort of the Lockheed-Georgia Company and Texas A&M University (TAMU) to develop an inverse wing design method that incorporates the latest advances in computational transonic aerodynamics. The research program includes three major tasks:

1. Formulation of the Design Method.
2. Development of a Three-Dimensional Inverse Pilot Code.
3. Development of a Baseline Unified Design/Analysis Code.

The formulation of the inverse design scheme and the development of a pilot code to validate the design method was conducted at Texas A&M University. This pilot code is based on the ZEBRA II three-dimensional transonic potential flow code developed at NASA Langley by South et al.[1-2] The development of the baseline unified design/analysis code was performed

by Lockheed-Georgia Company in two stages. A three-dimensional analysis code capable of solving the flow about swept, tapered wings without twist was developed first. This code served as the baseline code for the development of the unified design/analysis code. The inverse design method developed at TAMU was then implemented into this analysis code.

FORMULATION OF THE DESIGN METHOD

The principal goal of this research program was to develop a design method that is accurate, fast, and economical to use. To meet this goal, it was decided that the design method should incorporate the following features:

1. The inverse scheme would be based on the direct/inverse approach developed by Carlson at TAMU for a transonic airfoil design.
2. The potential flow solver would use the conservative form of the full potential equation.
3. Wing surface boundary conditions would be applied on a mean plane in a Cartesian grid system to simplify grid generation and wing shape calculation.
4. A fast, vectorizable solution algorithm such as approximate factorization or the ZEBRA II algorithm would be used in the potential flow solver.

In the direct/inverse design method, the leading edge geometry of the airfoil (usually the forward 10 percent) is specified; the remaining portion of the airfoil is computed for a specified pressure distribution. This eliminates the need to specify a boundary condition in the leading edge stagnation region.

The accuracy of transonic flow solutions for arbitrary swept wings depends on the form of the governing equation (i.e., full potential or small disturbance), the finite difference scheme, and the computational mesh system employed in the solution algorithm. The conservative form of the full potential equation provides the most accurate solution for highly swept wings. In addition, solution of the conservative full potential

equation ensures that the condition of zero mass flux across surface streamlines will be satisfied for inverse design cases. This condition is critical for the accurate calculation of wing shape.

The selection of a Cartesian computational grid system in lieu of a body-fitted system such as those used by Jameson[3] and Holst[4-6] was based on results obtained by Purcell and Carlson[7] for two-dimensional transonic flow. Purcell and Carlson[7] showed that sufficient accuracy can be obtained for full potential equation solutions by applying the full surface boundary condition on a mean plane in a Cartesian grid system. This plane can be located on a grid line or situated between two adjacent grid lines. Two problems are avoided by using the Cartesian grid system and mean plane boundary conditions for inverse design calculations. First, the computational grid does not have to be recomputed each time the wing shape is computed in an inverse design case. Second, intermediate calculation of wing shape during the potential flow solution is avoided. The new wing shape is computed only after the potential flow solution has converged to a desired value.

In order for any inverse scheme to be cost effective, the potential flow solver must be fast and reliable. In addition, the solution algorithm should be amenable to vectorization for use on current supercomputers such as the CYBER 205 and CRAY I. Two existing algorithms meet these requirements: the AF2 scheme developed by Holst[4-6] and the ZEBRA II algorithm developed by South et al.[1-2] After an unsuccessful attempt to implement the AF2 algorithm using the Cartesian grid system described, the ZEBRA II scheme was selected for the potential flow solver. The selection of the ZEBRA II scheme proved fortuitous because it allowed the use of the pilot code developed at NASA Langley as a test bed for developing the inverse design scheme. Since this code also used a Cartesian mesh system, the inverse schemes developed at TAMU using the ZEBRA II code could be implemented directly into the baseline analysis/design code being developed at Lockheed.

DEVELOPMENT OF THE THREE-DIMENSIONAL INVERSE PILOT CODE

The development of the inverse design method occurred in two phases. In the first phase, a scheme based on the small disturbance approximation to the surface boundary condition was developed. In the second phase, this technique was extended to use the full surface boundary condition applied on a mean plane in the Cartesian grid system. The small disturbance code was developed first because the NASA ZEBRA II code used the small disturbance approximation of the surface boundary condition. In addition, it was felt that the schemes developed for the small disturbance boundary condition provided a logical foundation for building the full potential scheme. Because of its importance in the development of the inverse design scheme, the ZEBRA II algorithm is described in detail in Appendix A. A complete description of the development of the inverse method is given in Reference 8.

SMALL DISTURBANCE DESIGN METHOD

The characteristics of the inverse design method in a Cartesian grid system are affected by the placement of the Z=0 plane on which the surface boundary conditions are applied. In the Langley ZEBRA II code the Z=0 plane is located between two grid lines. The surface boundary condition is implemented by replacing the difference approximation for ϕ_z in Eq. (A-3) of Appendix A on the plane KWNGT-1/2 or KWNGB+1/2 as shown in Figure 1, with the small disturbance approximation

$$\phi_z = U_\infty \frac{dz}{dx} \tag{1}$$

In this way, the surface boundary conditions can be implemented in the solution algorithm without using dummy values of potential or costly interpolations from the actual body surface. In addition, the complexity of the computer program is reduced.

In the inverse design method, the Neumann surface boundary condition is replaced by a Dirichlet condition in which the potential is specified directly as a function of a desired pressure distribution. In the present research, procedures were developed for implementing the inverse boundary

conditions that would not destabilize the convergence of the potential flow solution and would, at the same time, maintain the features of the ZEBRA algorithm that make it vectorizable. The following scheme was developed for the Cartesian grid system used in this research.

Referring to Figure 1, the small disturbance approximation of the pressure coefficient, $Cp = -2\phi_x$, at the mid-cell point X on the mean plane, Z=0, can be written

$$C_P = -\frac{2}{\Delta X}(\phi_R - \phi_L) \tag{2}$$

The objective is to compute the value of ϕ at point A as a function of the pressure specified at point X. This is accomplished by first computing ϕ_R and ϕ_L by extrapolation from the points above the mean plane. Three point extrapolation yielded the most accurate results. Therefore, ϕ_R can be written as

$$\phi_R = A\phi_A + B\phi_B + C\phi_C \tag{3}$$

where for an evenly space grid

$$A = \frac{z_B z_C}{(z_A - z_B)(z_A - z_C)}$$

$$B = \frac{z_A z_C}{(z_B - z_A)(z_B - z_C)} \tag{4}$$

$$C = \frac{z_A z_B}{(z_C - z_A)(z_C - z_B)}$$

Substituting Eq.(3) and a similar one for ϕ_L into Eq. (2), and solving for ϕ_A yields

$$\phi_A = \phi_D - \frac{1}{A}\left[B(\phi_B - \phi_E) + C(\phi_C - \phi_F) + \frac{\Delta x}{2} C_{R_X}\right] \tag{5}$$

When sweeping through the grid in the streamwise direction, the potential at the points corresponding to Point A at each inverse station in the cross-plane is determined using the previously described approach. Notice that this approach uses "old" values at the i cross-plane and "new" values

at i-1. The cross-plane is then solved using the standard ZEBRA approach. Since the ZEBRA scheme solves for $\Delta\phi$ only, it does not know which points are inverse and which are direct. At the end of the double pass ZEBRA loop, all points are updated by $\Delta\phi$ including the points corresponding to point A. To correct point A an additional calculation is performed to get back to its boundary value, i.e. $\phi_A = (\phi_A + \Delta\phi) - \Delta\phi$. This approach is needed to retain the vectorization feature and associated efficiency of the ZEBRA scheme. The actual ZEBRA loop remains blind to whether the station is inverse or direct. The boundary condition alone is changed.

After a converged inverse solution is obtained, the airfoil shape can be determined by integrating the airfoil slopes obtained from the wing boundary condition, i.e.

$$\frac{dz}{dx} = \frac{\phi_z}{U_\infty} \tag{6}$$

where ϕ_z must be obtained from the inverse solution. The first attempts to compute ϕ_z at the wing slit used the ϕ values from the inverse solution and the three point extrapolation formulas. However, this procedure did not yield accurate slopes and led to erroneous airfoil ordinates.

A second approach was devised that used the finite difference approximation of the full potential equation to obtain the wing slopes. Expanding Eq. (A-3) at point A and solving for ϕ_z at R yields

$$\phi_{z_R} = \frac{1}{\rho_R}\left\{(\rho\phi_z)_B + \Delta z\left[\bar{\delta}_x(\rho\phi_x)_{i=A+\frac{1}{2}} + \bar{\delta}_y(\rho\phi_y)_{j=A+\frac{1}{2}}\right]\right\} \tag{7}$$

The value of wing slope at each inverse station is then obtained by substituting Eq. (7) for ϕ_z in Eq. (6). Straightforward trapezoidal integration is then used to obtain the airfoil ordinates at each inverse station, i.e.,

$$z_{i+1} = z_i + \frac{\Delta x}{2}\left[\left(\frac{dz}{dx}\right)_{i+1} + \left(\frac{dz}{dx}\right)_i\right] \tag{8}$$

FULL POTENTIAL DESIGN METHOD

As in the small disturbance method, the full potential design method uses a specified pressure coefficient to define an inverse boundary condition. However, the full potential pressure coefficient equation is used in place of the linearized small disturbance equation. The full potential pressure coefficient can be written as

$$C_p = \frac{2}{\gamma M_\infty^2}\left[1 + \frac{(\gamma - 1)}{2} M_\infty^2 \left(1 - \frac{u^2 + v^2 + w^2}{q_\infty^2}\right)^{\frac{\gamma}{\gamma-1}} - 1\right] \tag{9a}$$

where M_∞ is the freestream Mach number, γ is the ratio of specific heats, q_∞ is the magnitude of the freestream velocity vector, and u,v,w are the local components of velocity given by the expressions

$$\frac{u}{q_\infty} = 1 + \phi_x$$

$$\frac{v}{q_\infty} = \phi_y \tag{9b}$$

$$\frac{w}{q_\infty} = \phi_z$$

Solving Eq. (9) for ϕ_x yields

$$\phi_x = \left\{\frac{1 - \frac{2}{(\gamma - 1)M_\infty^2}\left[\left(1 + \frac{C_p\ \gamma\ M_\infty^2}{2}\right)^{\frac{\gamma-1}{\gamma}} - 1\right]}{1 + \left(\frac{v}{u}\right)^2 + \left(\frac{w}{u}\right)^2}\right\}^{\frac{1}{2}} - 1 \tag{10}$$

Referring to Figure 1, this expression is applied at the point X to extract a value for the potential at point A as was done in the small disturbance method. However, values of ϕ_y and ϕ_z must now be calculated. As in the small disturbance method a three point Lagrangian extrapolation is used to define values of ϕ on the wing mean plane.

The term ϕ_y is given in the half plane at Point X by the average of ϕ_y at points R and L, i.e.,

$$\phi_y = \frac{1}{2}\left(\frac{\phi_H - \phi_G}{2\Delta y} + \frac{\phi_N - \phi_M}{2\Delta y}\right) \tag{11}$$

The value of ϕ_z at point X is computed by averaging ϕ_z at L and R. The expression for ϕ_z in the half plane is given by differentiating the general three-point Lagrangian extrapolation with respect to Z so that ϕ_z at R is given by

$$\phi_z = \bar{A}\phi_A + \bar{B}\phi_B + \bar{C}\phi_C \tag{12}$$

where

$$\begin{aligned}
\bar{A} &= \frac{-z_B - z_C}{(z_A - z_B)(z_A - z_C)} \\
\bar{B} &= \frac{-z_A - z_C}{(z_B - z_A)(z_B - z_C)} \\
\bar{C} &= \frac{-z_A - z_B}{(z_C - z_A)(z_C - z_B)}
\end{aligned} \tag{13}$$

It can be seen, however, that evaluating ϕ_z and ϕ_y with these expressions poses several problems when used in Eq. (10). First, Eq. (10) cannot be evaluated explicitly at each time step since u appears in the denominator of the right-hand side of the equation. Second, ϕ_A appears in both ϕ_z and ϕ_x expressions. These problems were overcome by updating v/u and w/u only every ten iterations using the current value of ϕ_A. This procedure is based on the fact that ϕ_y and ϕ_z should be on the same order of

magnitude as the slopes and, therefore, very small relative to ϕ_x. Thus, the denominator in Eq. (10) should remain on the order of one in the inverse region.

As in the small disturbance method, the final expression for ϕ_A is given by

$$\phi_A = \phi_D + \frac{\Delta x}{A}\left\{\frac{1 - \frac{2}{(\gamma-1)M_\infty^2}\left[\left(1 + \frac{C_p\,\gamma M_\infty^2}{2}\right)^{\frac{\gamma-1}{\gamma}} - 1\right]}{1 + \left(\frac{v}{u}\right)^2 + \left(\frac{w}{u}\right)^2}\right\}^{\frac{1}{2}} - \frac{\Delta x}{A} \qquad (14)$$

$$- \frac{B}{A}(\phi_B - \phi_E) - \frac{C}{A}(\phi_C - \phi_F)$$

where A, B, and C are the Lagrangian coefficients given in Eq. (4). This boundary condition is implemented in the ZEBRA algorithm in the same manner as the small disturbance boundary condition. The ZEBRA code was also modified to use the full surface boundary condition applied on the wing mean plane for calculations in the direct region.

Calculation of wing slopes in the full potential method is also performed in the same manner as was done in the small disturbance method by using the residual equation to define the w velocity on the wing mean plane. However, a modified form of the full surface boundary condition is used in place of the small disturbance condition. The full potential boundary condition can be written as

$$\phi_z = (1 + \phi_x)\frac{dz}{dx} + (\phi_y)\frac{dz}{dy} \qquad (15)$$

In the current design method, the spanwise slope is set to zero. As in the small disturbance method, ϕ_z is computed using Eq. 7 after the scheme has iterated to a desired level of convergence.

INVERSE METHOD VERIFICATION

The verification test centered on validating the design consistency and accuracy of both inverse methods. The design consistency means that a wing shape generated for a given pressure distribution in the inverse mode will yield the same pressure distribution when run in a purely analysis mode. A test of both consistency and accuracy is to take the pressure distribution for a known wing shape as the target pressure distribution for an inverse design and then compare the computed shape with the original wing. Both techniques were used in the present research.

Small Disturbance Method

Both the small disturbance and the full potential design methods were verified using an untapered NACA 0012 wing with an aspect ratio of 6.96. A planform view of this wing configuration indicating the span stations used in the design testing is shown in Figure 2. Tests were made for both subcritical and supercritical Mach numbers. All the tests were made using a 72x17x30 grid.

Figures 3 and 4 present results for the small disturbance design scheme at a single span station at two angles of attack. These figures compare the computed pressures and target pressures for two subcritical tests. In Figure 3, the input pressures for the inverse scheme were the same as those obtained from analysis for a 2 degree angle of attack. As can be seen, the resulting pressures obtained at the end of the inverse cycle compare well with the analysis pressures.

Figure 4 presents results for a test at zero angle of attack using a modified upper surface pressure distribution. The pressure distribution obtained at the end of the inverse cycle is compared with the target pressure distribution. The pressures in the inverse region compare quite well. However, the pressures on the lower surface and in the nose region of the upper surface are slightly changed. These changes are to be expected since the lift and, therefore, the circulation of the wing has been changed.

The results obtained for the small disturbance scheme indicated that the method was convergent and duplicable for a given pressure distribution. Based on these results, the method was extended to the full potential scheme. A series of tests were run in which the inverse scheme was applied at three consecutive span stations.

Subcritical Tests

The first runs with the full potential code were made to test the accuracy of the code by using the pressure distribution for a known airfoil shape as the target for the inverse mode, and then comparing the resulting shape with the original airfoil. Figures 5 to 7 present results at a single span station for a subcritical test at 2 degrees angle of attack. Figure 5 demonstrates that the full potential method will accurately reproduce the desired pressure distribution. The accuracy of the inverse scheme and the shape calculation is shown by Figures 6 and 7. As can be seen, both the NACA 0012 slopes and ordinates are accurately computed.

Figures 8 through 19 present results at all three design stations for a subcritical test at zero angle of attack for a modified upper surface pressure distribution. The target pressure distribution was obtained by modifying the analysis pressures with a french curve. It was not expected that specifying the target pressure distribution in this arbitrary manner would produce realistic airfoil shapes. However, the object of these tests was to determine if the specified pressure distribution could be reproduced by the inverse scheme.

Figures 8 to 10 present the pressure distribution for all three span stations. As previous tests, the computed pressures agree well with the specified pressures in the design region with small changes evident on the lower surface and the nose region of the upper surface. The airfoil slopes and ordinates computed for the specified design pressures are given in Figures 11 through 16. Note that these new airfoils all have "fishtails," i.e., the upper and lower surfaces cross, which is physically unrealistic. However, these results illustrate one of the problems encountered in developing an inverse scheme. Because the purpose of these tests was to verify

the inverse approach and not to design a wing, no attempt was made to control the trailing edge thickness.

Figures 17 through 19 illustrate the consistency of the design scheme. The pressures generated by analyzing the modified wing shape are compared with the pressures obtained from the inverse code. These results indicate that the modified wing shapes will yield the desired pressures when the wing is analyzed.

Supercritical Tests

The next series of tests were made to verify the inverse scheme for supercritical flow. Following the same procedures used for subcritical flow, the code was first tested for accuracy by using a pressure distribution obtained from analysis of the wing, as the target pressure distribution for the inverse scheme. Figure 20 to 22 presents results for the unswept NACA 0012 wing at a 2 degree angle of attack and a Mach number of 0.82. As in the subcritical cases, the pressures are in excellent agreement.

The next step in the supercritical testing was to modify the Mach 0.82 two-degree angle of attack pressure distribution to eliminate the shock on the upper surface. This type of design represents a typical application of an inverse scheme in a wing design. Figures 23 to 25 compare the pressure distributions obtained by the inverse scheme with the desired pressure distributions at the three design stations. These results further confirm that the inverse scheme is successful at supercritical Mach numbers.

Figures 26 through 31 show the corresponding airfoil slopes and shapes obtained from the inverse code. Note that for this design the trailing edge thicknesses of the resulting airfoils are physically unrealistic. However, Figure 32 illustrates that these shapes are consistent with the specified pressure distribution. The results at the other two design stations compare equally well.

Swept Wings

A final series of tests was conducted to verify the inverse method for swept wings. The supercritical tests were repeated using the basic NACA 0012 wing swept back 15 degrees. The Mach number for the tests was increased to 0.85. The results presented are for an angle of attack of zero degrees. Figures 33 to 35 compare the computed and specified pressured distributions for a test in which the upper surface pressures were modified to eliminate the shock. The airfoil slopes and shapes generated by the modified pressure distribution are given in Figures 36 through 38. These figures show that the accuracy of the code is not affected by wing sweep. Figures 42 to 44 show that the consistency of the design scheme is retained.

All the test cases described were converged to a residual of 0.0011, which represents a reduction from the initial residual of about three orders of magnitude. For most cases, this represents a sufficient level of convergence. This convergence criteria required an average of about 83 seconds on a Cyber 203 for inverse runs and about 63 seconds for analysis runs.

DEVELOPMENT OF THE BASELINE UNIFIED DESIGN/ANALYSIS CODE

The baseline unified design/analysis code was developed in two stages. In the first stage, a three-dimensional transonic flow analysis code was developed that served as the foundation for the development of the baseline design/analysis code. In the second stage, the inverse design method developed at Texas A&M University was implemented.

TRANSONIC POTENTIAL FLOW ANALYSIS

A new analysis code had to be written because the NASA Langley ZEBRA II code was developed as a pilot code to verify the ZEBRA algorithm and was, therefore, not suited for calculating flow about arbitrary wing configurations. Additionally, the ZEBRA II code (and consequently, the TAMU pilot code) was written in explicit vector instructions which prohibits its

use on computers other than the CYBER 203 and 205. The baseline unified design/analysis code was written in standard FORTRAN to make it transportable to other computers. The major difference between the baseline code and the TAMU code is the use of streamwise shearing transformations to align the computational mesh with the wing planform.

Grid Generation

Prior to the start of this research, it was felt that the fine inner mean/coarse outer mesh grid embedding scheme developed by Boppe[9] would provide the optimum mesh system for the design code. However, after work on the design method began, it was decided that the time required to implement the embedded grid system would delay the development of the unified analysis/design method. Therefore, both analysis and inverse solutions were obtained using one mesh for the entire computational domain. Initial tests were made using a 90x30x30 grid. Because the convergence rate for this grid system was unacceptably slow, grid sequencing was employed to speed up convergence.

The computational grid for the baseline analysis code was formed by first computing a stretched cartesian grid system and then shearing the grid to align it with the leading and trailing edges of the wing. This procedure has been used with great success in the small-disturbance codes developed by Bailey and Ballhaus[10] and by Boppe[9]. With the sheared grid system, each spanwise plane of the grid contains an equal number of points on or adjacent to the wing surface.

To ensure compatibility with the TAMU pilot code, the grid stretching used in that code was implemented in the Lockheed code. In this mesh system, the wing surface is covered with an evenly spaced grid system. The regions in front of the leading edge, behind the trailing edge, and outboard of the wing tip are stretched exponentially. Geometric stretching is used above and below the wing mean plane.

A shearing transformation of the form

$$X(x,y) = \frac{x-xle(y)}{c(y)}$$

$$Y(y) = y \qquad (16)$$

$$Z(z) = z$$

(where xle(y) defines the leading edge of the wing and c(y) is the local chord distribution), transforms the physical grid system (x,y,z) into a computational grid (X,Y,Z) aligned with the wing. Figure 45 shows the mesh system generated by this transformation.

In a typical inverse or analysis solution, a sequence of three grids is used. The solution starts on a coarse grid that has 25 chordwise points, 30 spanwise points, and 8 points normal to the wing mean plane. The solution from the coarse grid is interpolated onto a medium grid that contains 50 chordwise points, 30 spanwise points,and 16 normal points. The medium grid solution is interpolated onto a fine grid that has 90 chordwise points, 30 spanwise points, and 30 points normal to the mean plane. Twenty of the 30 spanwise stations used in each grid were placed on the wing surface. The number of chordwise points covering the local chord at each span station varied from 10 for the coarse grid to 25 for the medium grid and 50 for the fine grid.

The Full Potential Equation in General Coordinates

Following Holst[4-6] a general coordinate transformation is used to transform Eq. (A-1) to the computational coordinate system. On transformation, Eq. (A-1) becomes

$$\left(\frac{\rho U}{J}\right)_X + \left(\frac{\rho V}{J}\right)_Y + \left(\frac{\rho W}{J}\right)_Z = 0 \qquad (17a)$$

$$\rho = \left[1 - \frac{\gamma - 1}{\gamma + 1}\left(U\phi_X + V\phi_Y + W\phi_Z\right)\right]^{\frac{1}{\gamma - 1}} \qquad (17b)$$

where U, V, and W are the contravariant components of velocity in the computational plane and J is the Jacobian of the transformation. Details of this transformation are given in References 4 to 6. Eq. (17) has been nondimensionalized by the critical speed of sound and the stagnation density.

For the transformation defined by Eq. (16), we get

$$\begin{aligned} U &= (X_x^2 + X_y^2)\ \phi_X + X_y\ \phi_Y \\ V &= X_y\ \phi_X + \phi_Y \\ W &= \phi_Z \\ J &= X_x \end{aligned} \tag{18}$$

This transformation retains the strong conservation form of the original equation.

Numerical Solution Algorithm

The finite difference analog of Eq. (17) can be written

$$\bar{\delta}_X \left(\frac{\rho U}{J}\right)_{i+\frac{1}{2},j,k} + \bar{\delta}_Y \left(\frac{\rho V}{J}\right)_{i,j+\frac{1}{2},k} + \bar{\delta}_Z \left(\frac{\rho W}{J}\right)_{i,j,k+\frac{1}{2}} = 0 \tag{19}$$

where $\bar{\delta}_X$, $\bar{\delta}_Y$ and $\bar{\delta}_Z$ are first order backwards differences. In order to maintain stability in regions of supersonic flow, the density has been replaced by the retarded density approximation used by Holst.[4-6]

$$\bar{\rho}_{i+\frac{1}{2},j,k} = [(1-\nu)\rho]_{i+\frac{1}{2},j,k} + \nu_{i+\frac{1}{2},j,k}\rho_{i-\frac{1}{2},j,k} \tag{20a}$$

where

$$\nu = \text{MIN}\ [1,\ \text{MAX}\ (1 - \frac{M_c}{M^2_{i,j,k}},\ 0)] \tag{20b}$$

and $\bar{\rho}$ is the retarded density coefficient; M_c is a cutoff Mach number whose value is usually $0.94 \leq M_c \leq 1.0$.

In the current code, Eq. (20) is evaluated only at the midsegment point i+1/2,j,k. The values at i,j+1/2,k and i,j,k+1/2 are obtained by averages of the surrounding points. Values on the mean plane are obtained by two point extrapolation from above and below the mean plane. Averages of central difference approximations are used to compute the values of ϕ_y at i±1/2 and ϕ_x at J±1/2 in Eq. 18.

It was found that the convergence of the analysis method was improved by splitting the full potential into separate perturbation and freestream components.

$$\phi = G + X\, q_\infty \cos \alpha + Z q_\infty \sin \alpha \qquad (21)$$

where α is the angle of attack. Equation (19) is then solved for G.

Boundary Conditions

As in the NASA Langley ZEBRA code, surface boundary conditions are introduced by replacing W at KWNGB+1/2 and KWNGT-1/2 in Eq. (19) with Eq. (15). This boundary condition is computed prior to the start of each iteration using values of ϕ from the previous iteration and then held constant during the ZEBRA sweeps.

For lifting cases, the section circulation Γ is computed by taking the difference in the potential at the section trailing edge linearly extrapolated from the points above and below the airfoil. The Kutta condition is implemented in Eq. (19) by replacing ϕ_{KWNGB} with $\phi_{KWNGB}-\Gamma$ on the line above the mean plane and by replacing ϕ_{KWNGT} with $\phi_{KWNGT}+\Gamma$ on the line below the wing plane at all points behind the trailing edge of the wing. The circulation is computed prior to the start of each iteration and then slightly overrelaxed to improve convergence.

Because the outer boundaries are located at finite lengths from the wing, the expressions of Klunker[11] are used to compute the change in the far-field potential due to lift. The potential on the downstream boundary is updated by assumming that $G_x = 0$. This was implemented by letting $G_{NI} = G_{NI-1}$.

The symmetry condition at the wing root is implemented by setting $V = 0$ and by replacing ϕ_Y in Eq. (18) with

$$\phi_Y = - X_y \phi_X \tag{22}$$

Verification of the Analysis Code

A standard wing used to evaluate the performance of an analysis code is the ONERA M6 wing described in Reference 12. This wing has a leading edge sweep angle of 30 degrees, an aspect ratio of 3.8, and a taper ratio of 0.562. A series of verification tests were conducted at a Mach number of 0.839 and an angle of attack of 3.0 degrees. Figures 46 to 49 compare the computed pressure distribution at four span stations on the wing with experimental data and results from TWING program of Holst. As can be seen, the results for this test case compare reasonably well with both the experimental data and the TWING results. The discrepancies at the wing tip and at the lower-surface leading edge indicate the need for finer grid systems in these regions. The pressure distribution over the entire wing is given in Figure 50.

In a typical analysis run, the coarse and medium grids are iterated until the initial residual drops by four orders of magnitude or a maximum number of iterations, usually 400, is reached. The code is then run for 100 iterations on the fine grid. The current version of the baseline code takes about 90 seconds of CPU time on a CRAY 1S computer and about 8 minutes of CPU time on the CYBER 203 to run the ONERA M6 test case. This disparity in run times points out the inefficiency of the automatic vectorization feature of the CYBER 203 FORTRAN compiler.

INVERSE METHOD

The inverse method developed at Texas A&M University was implemented in the same manner as was done in the ZEBRA II pilot code. However, the non-dimensional form of the governing equations used in the Lockheed code leads to a slightly different equation for the pressure coefficient boundary condition. The derivation of this equation is given in Appendix B.

A series of tests were initiated to determine the consistency and accuracy of the inverse method using the ONERA M6 wing as the base geometry. First, the pressure distribution for the ONERA M6 wing generated by analysis for the Mach 0.84 case was used as the target pressure distribution for the inverse code. The resulting pressure distribution and wing shapes were then compared with the corresponding data for the base shape. The inverse method was applied at five consecutive design stations along the span. Figures 51 to 53 compare the base airfoil sections with the sections computed by the inverse scheme. The airfoil shapes are recovered reasonably well. The error in the shape calculation is felt to be due to neglecting the effects of the spanwise variation in slope in the inverse scheme for a tapered wing. In addition, the target pressures were generated using the full potential boundary condition. Techniques to include the spanwise effects in the inverse scheme will be investigated in the second phase of this contract.

Next, the inverse scheme was tested using target pressure distributions designed to weaken the shock at each of the five design stations. Figures 54 to 56 compare the computed pressure distributions with the target pressure distributions at three of the design stations. The computed and the target pressures are in good agreement. Figures 57 to 59 compare the base airfoils with the computed airfoils by the inverse method. As in the results obtained by the TAMU pilot code, the airfoils produced are physically unrealistic. Figures 60 to 62 indicate, however, that the computed wing shape is consistent with the specified pressure distribution.

In a typical inverse run, the coarse grid is converged for the base wing geometry. The coarse grid results are interpolated onto the medium grid and run for 50 iterations before the inverse design is initiated. The medium grid is then run in the inverse mode until convergence. The medium grid results are interpolated onto the fine grid which is then run for the 100 iterations in the inverse mode. Wing shapes are computed at the end of the fine grid iterations. Running the code in the inverse mode requires only about 100 seconds on the CRAY 1S.

CONTROL OF TRAILING EDGE THICKNESS

The results from both the baseline and pilot inverse design codes indicate the need for a technique to control trailing edge thickness. A review of existing wing and airfoil inverse design methods revealed two different procedures for enforcing trailing edge closure that can be used in the present design method.

In the first approach, the nose region of the starting airfoil or wing is modified to increase or decrease the leading edge radius. After a few tries, a nose shape can usually be found that will provide the desired trailing edge thickness. This approach was used by Carlson[13-14] in his two-dimensional direct/inverse design method. The major drawback of this approach is that it is a trail and correction procedure that relies heavily on the expertise of the designer. Shankar[15-16] has suggested that this procedure can be automated by specifying the nose shape by $y = a_o x^n$, where n and a_o are free parameters that are adjusted by a numerical optimization procedure to satisfy a specified trailing edge thickness constraint.

A second technique to enforce trailing edge closure has been used by Shankar[16] in a small-disturbance design code. In this technique, a functional relationship between the trailing edge thickness and the velocity potential at the leading edge is assumed at each spanwise design station. A perturbation to the leading edge velocity that will drive the trailing edge thickness to a desired value can then be computed at each design station.

At the start of this contract it was felt that perturbing the nose shape under control of a numerical optimization scheme would prove to be the most effective approach for enforcing trailing edge closure. Therefore, the baseline unified design/analysis code was modified to use the optimization techniques developed by Vanderplaats, Hicks, and Murman[17] to perform the nose shape modifications required to control trailing edge thickness. However, the time constraints on the first phase, of the contract prevented the successful implementation of optimization. The best way to control the trailing thickness will be addressed in the second phase of the contract.

RESULTS AND DISCUSSION

The results presented in the previous sections of this report illustrate that the present design method can be used effectively for both subcritical and supercritical design cases. The supercritical results show that the method can perform one of the more important functions of a transonic wing design code--the elimination or weakening of strong shock waves at supercritical Mach numbers. This can be seen by comparing the section lift, drag, and moment coefficients shown in Fig. 48 for the 67% span station of the base ONERA wing with the values given in Fig. 62 for modified wing. The lift coefficient changes from .21 for the base wing to .020 for the modified wing. The drag coefficient changes from -0.0081 to -0.001 and the moment coefficient changes from -0.0960 to -0.0947. The negative values of drag illustrate the inaccuracy of the pressure drag calculation that is common to most transonic analysis codes. However, the values obtained illustrate the effect that weakening the shock has on the section drag.

CONCLUSIONS

An inverse design method for wings in transonic flow that is particularly suited for use on a vector computer has been developed. The technique has been verified for accuracy and consistency in both a developmental pilot code and a baseline unified/design code that will serve as the basis for future code development.

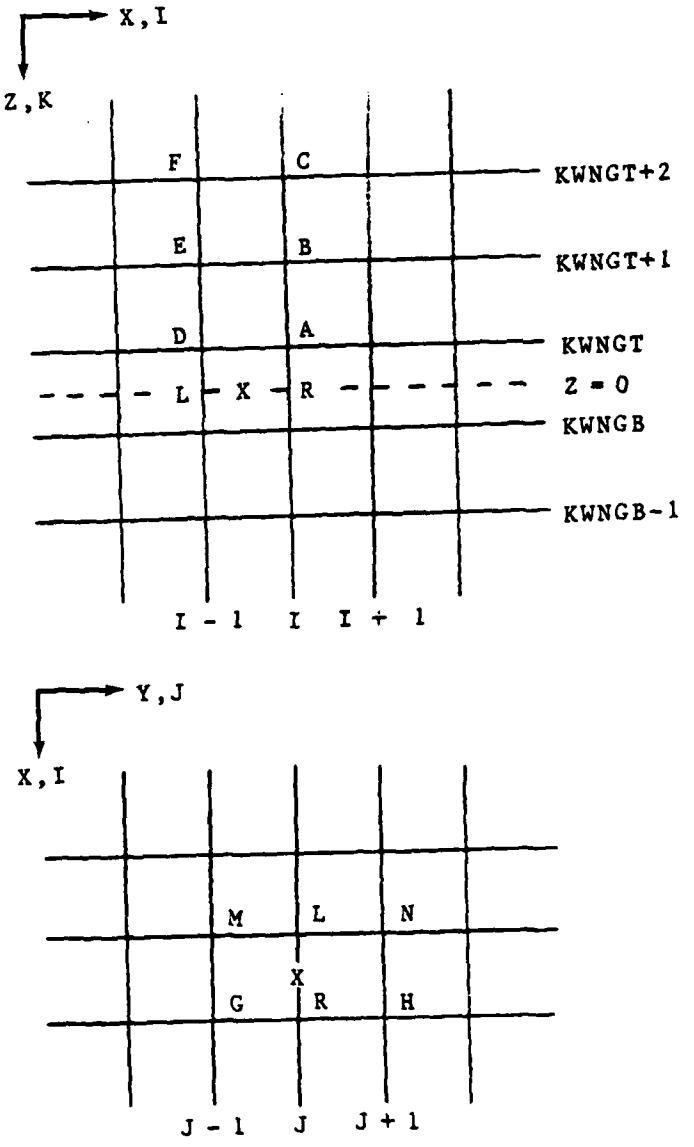


Figure 1. Grid points used in inverse boundary condition calculation

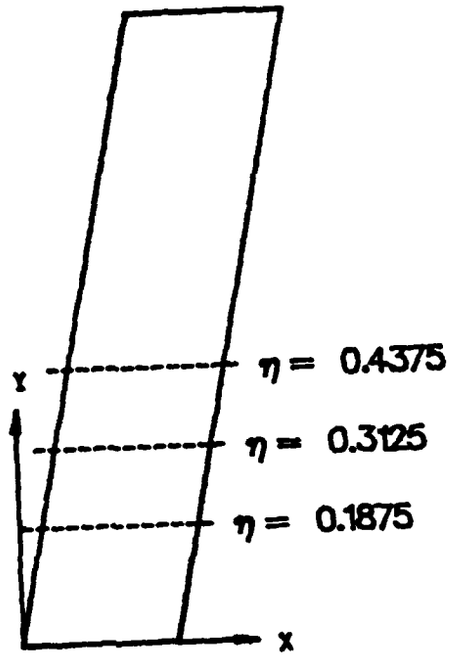


Figure 2. Design stations used in TAMU pilot code

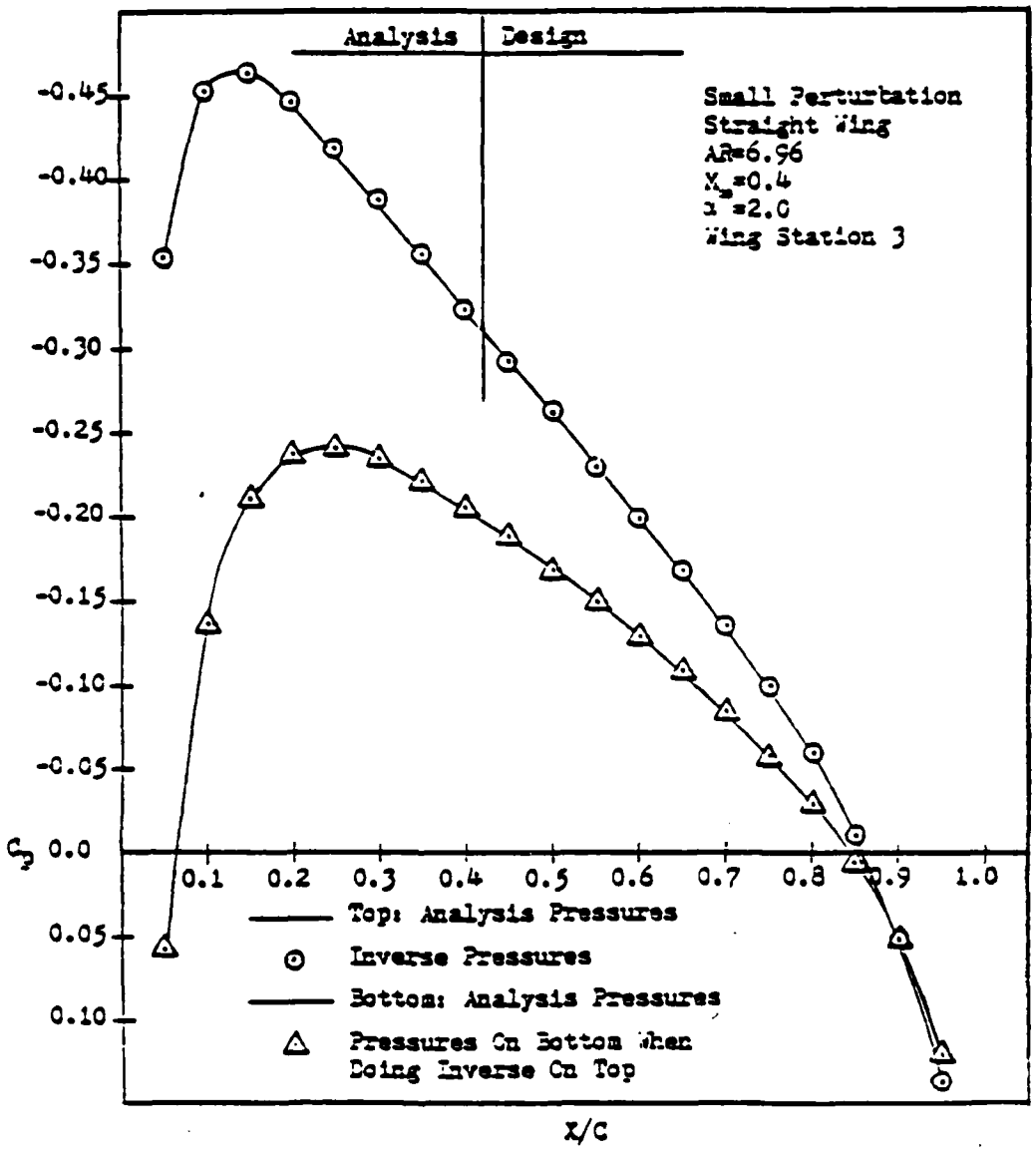


Figure 3. Correlation of small perturbation inverse pressures with analysis target pressures

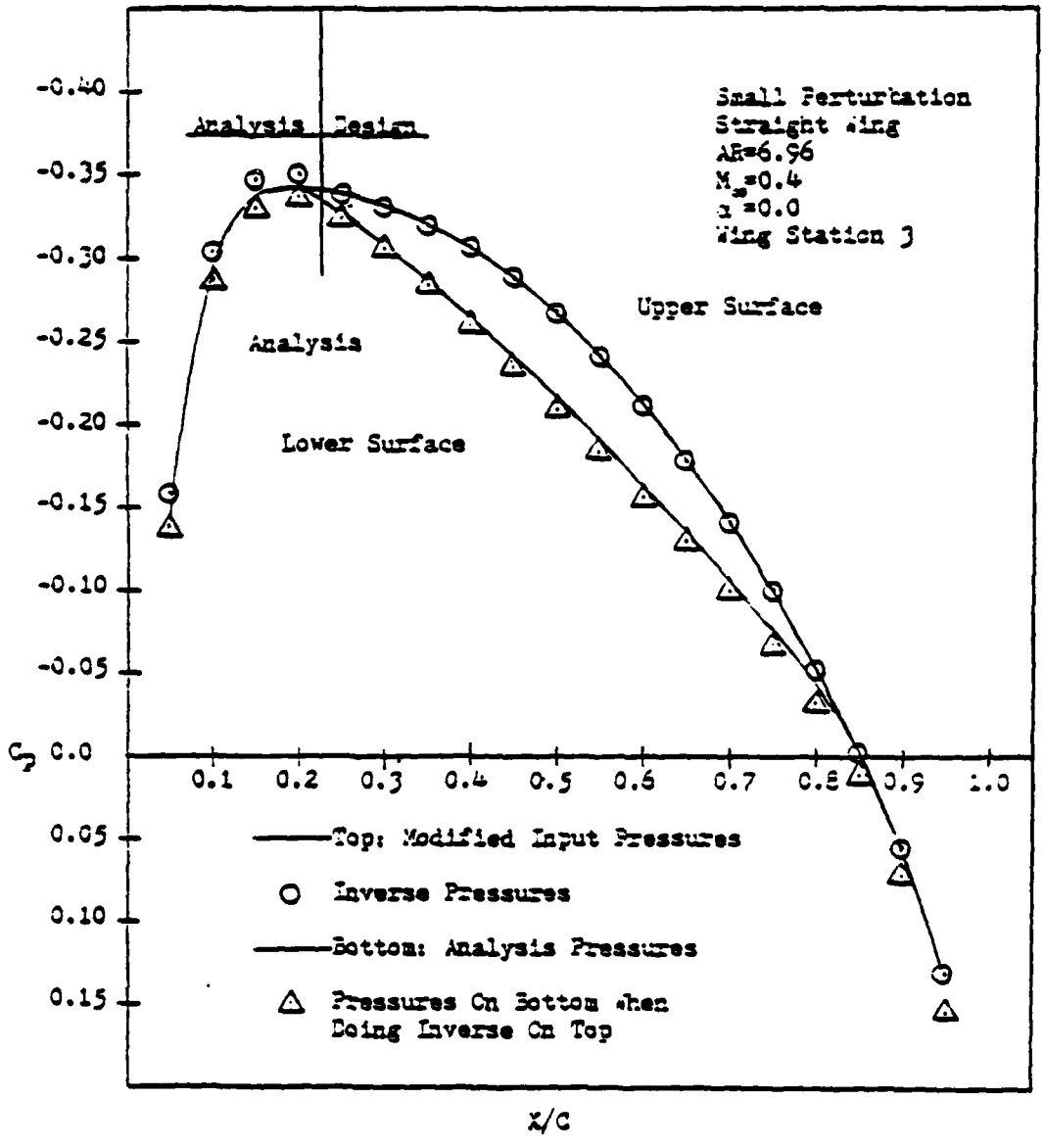


Figure 4. Correlation of small perturbation inverse pressures with modified target pressures

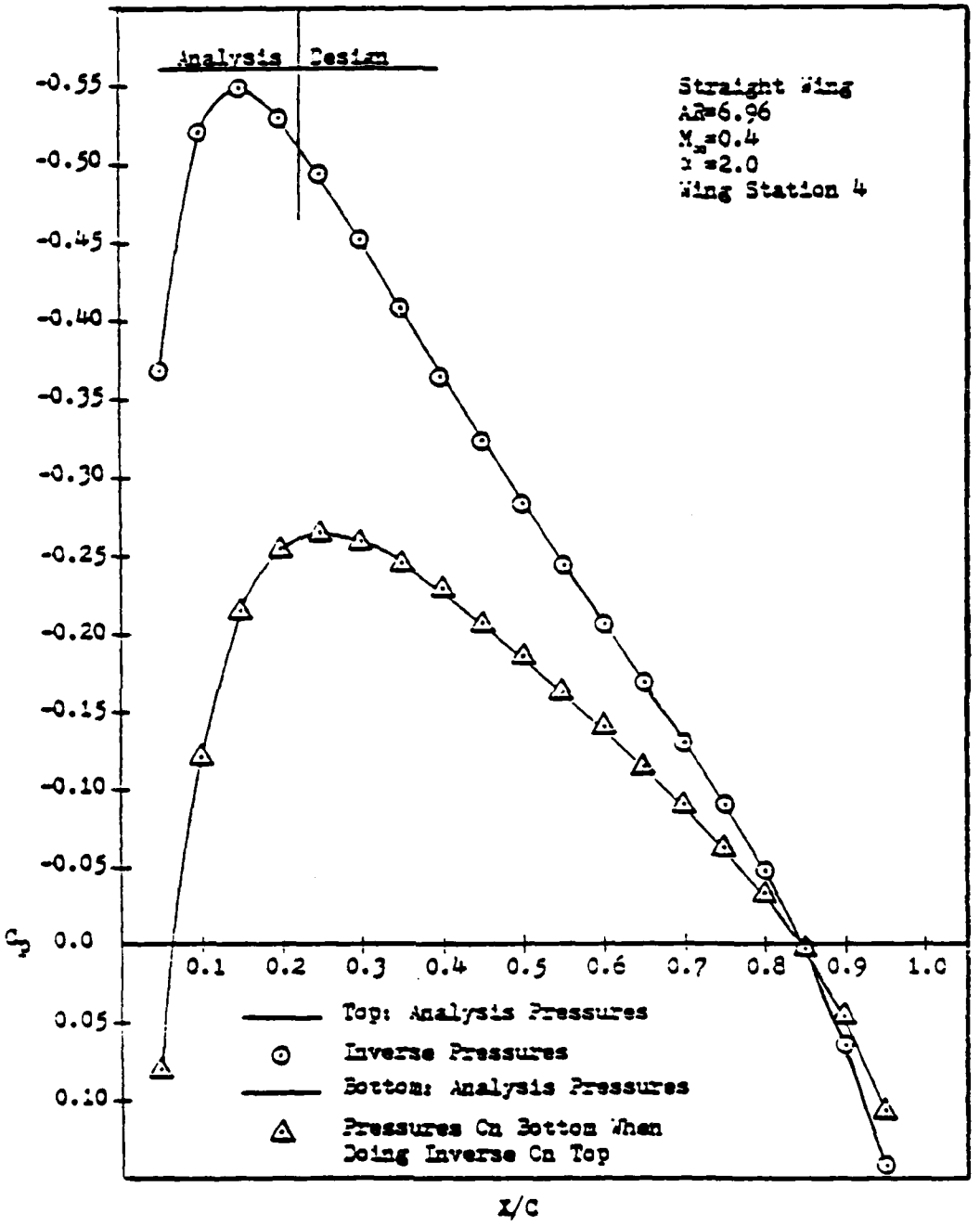


Figure 5. Correlation of full potential inverse and analysis target pressures, Mach = 0.4, η = 0.3125

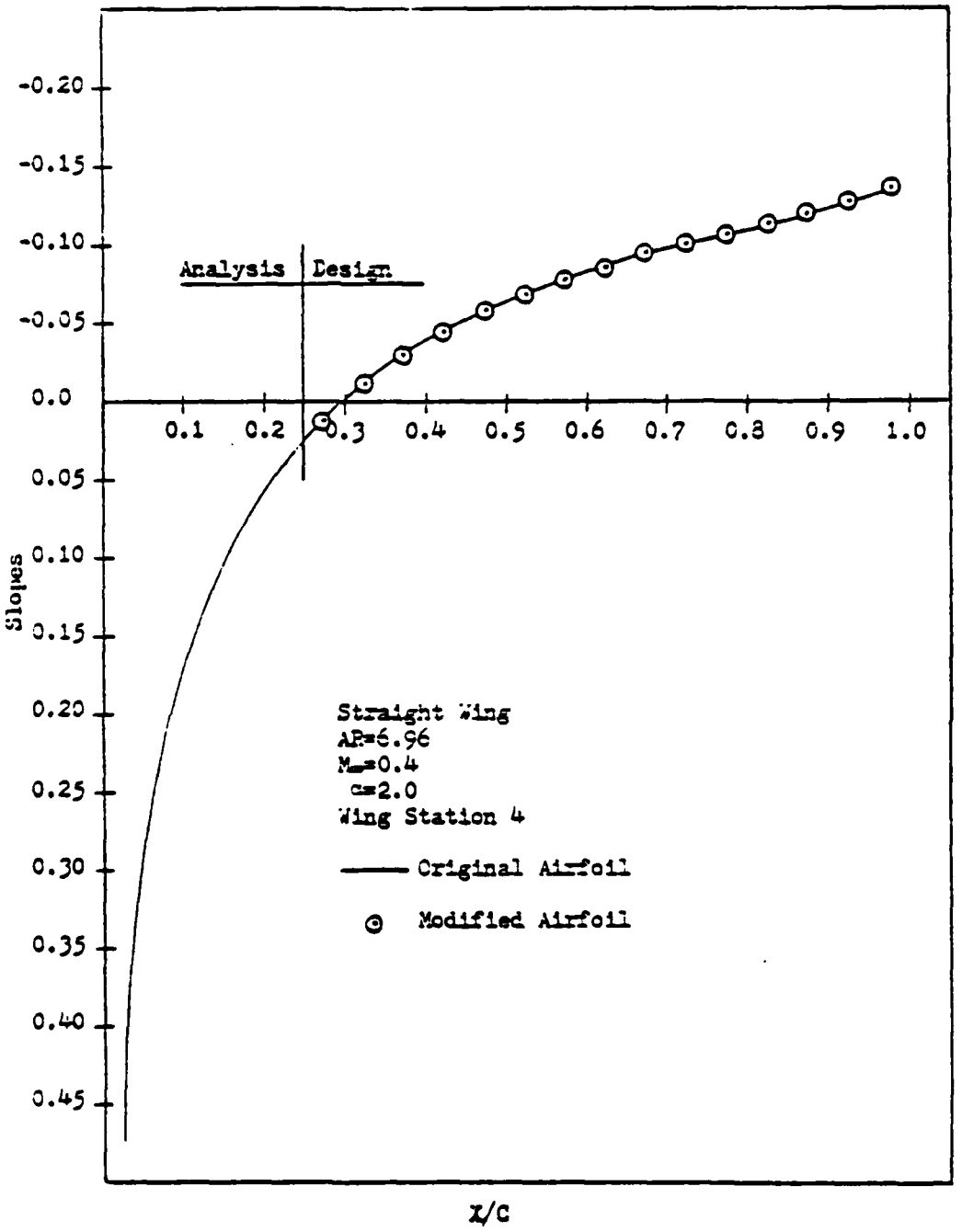


Figure 6. Correlation of original airfoil slopes with slopes from inverse using analysis target pressures

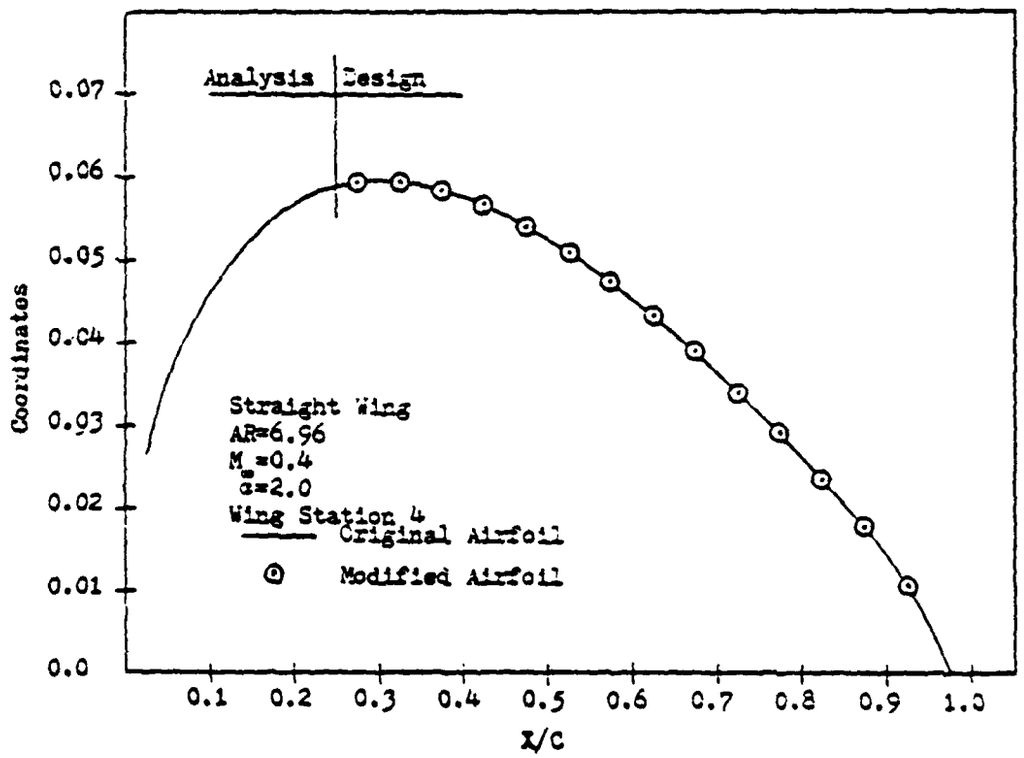


Figure 7. Correlation of original airfoil ordinates with ordinates from inverse using analysis target pressures

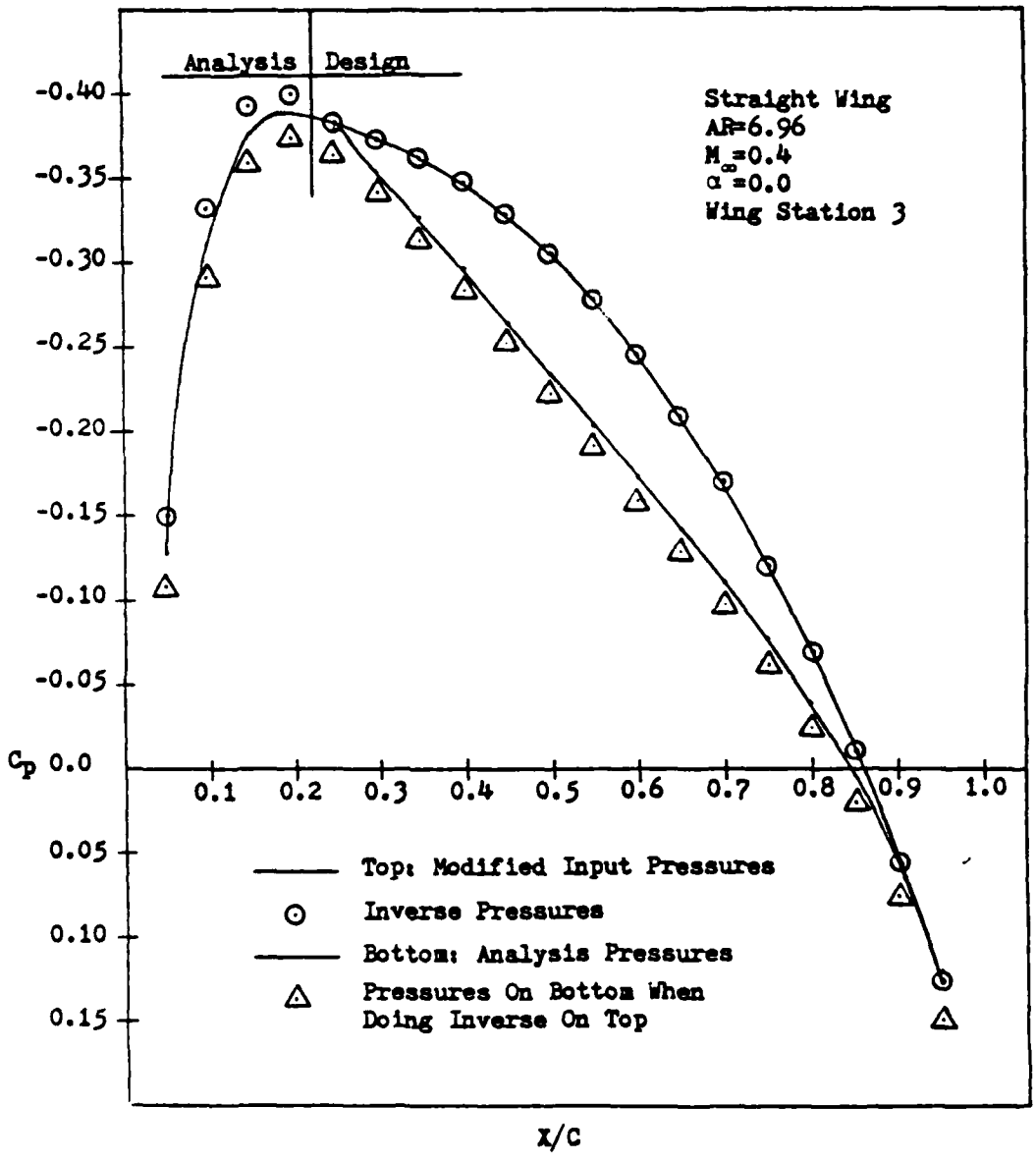


Figure 8. Comparison of inverse pressures and modified target pressures, Mach = 0.4, η = 0.1875

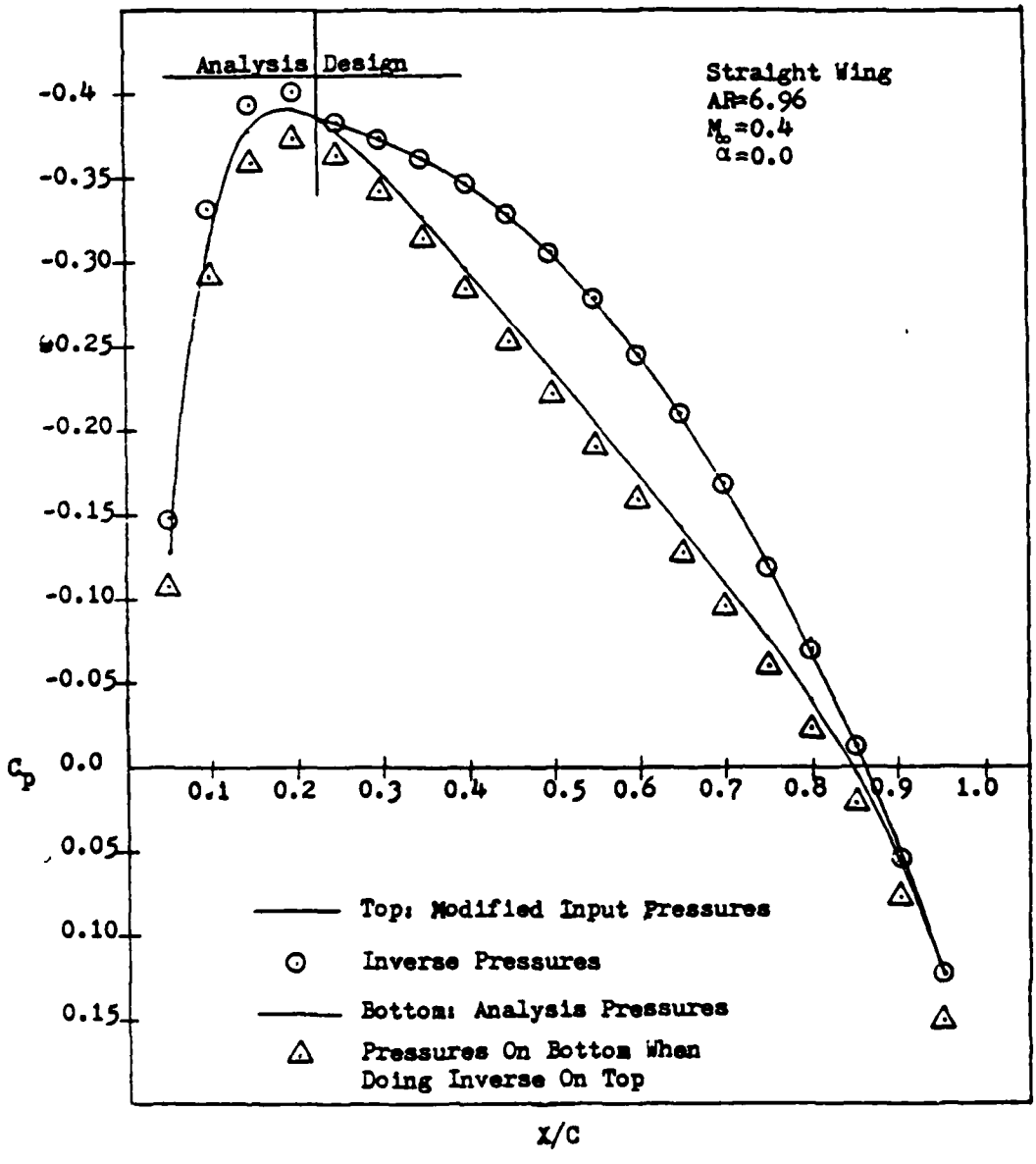


Figure 9. Comparison of inverse pressures and modified target pressures, Mach = 0.4, n = 0.3125

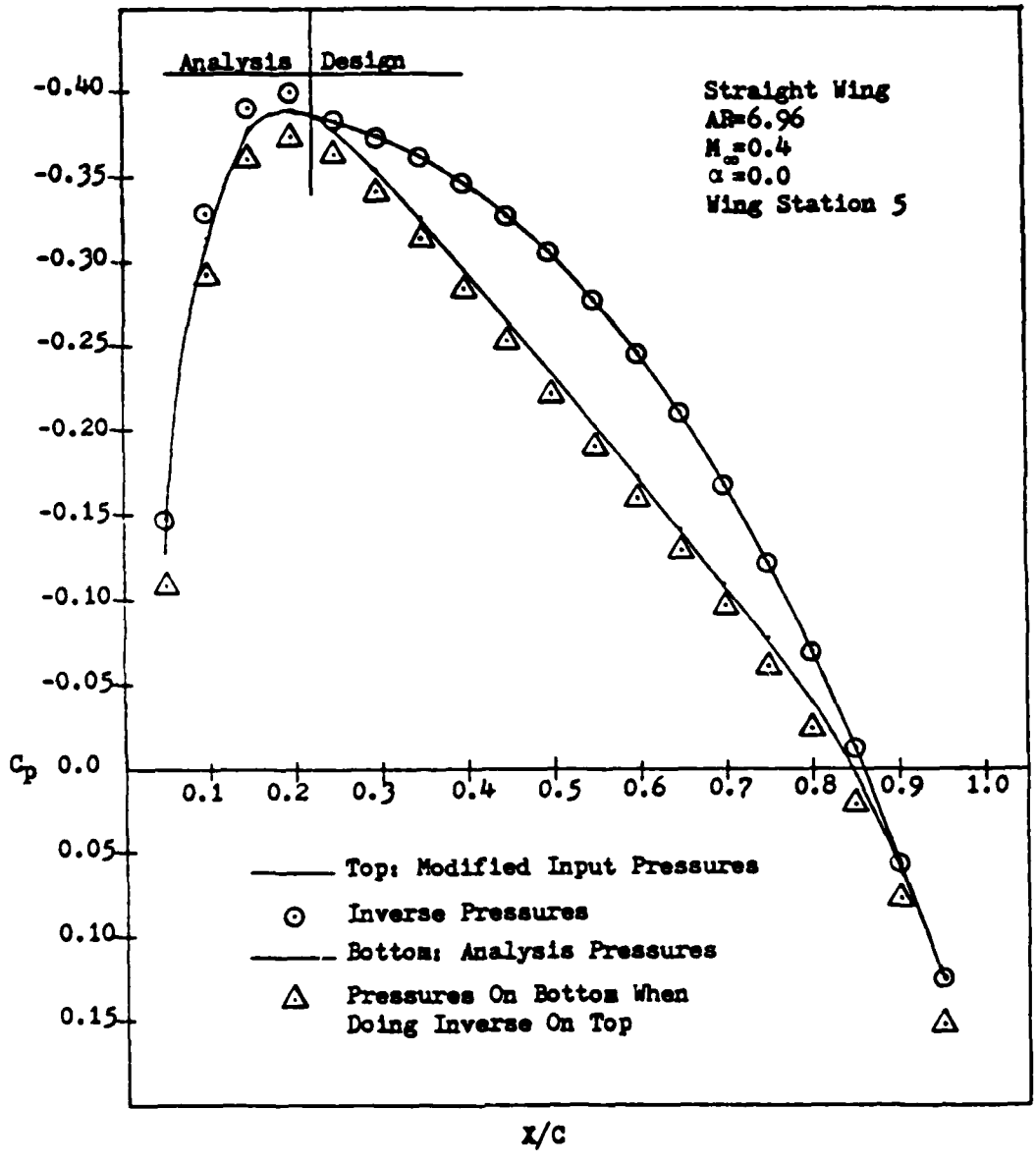


Figure 10. Comparison of inverse pressures and modified target pressures, Mach = 0.4, η = 0.4375

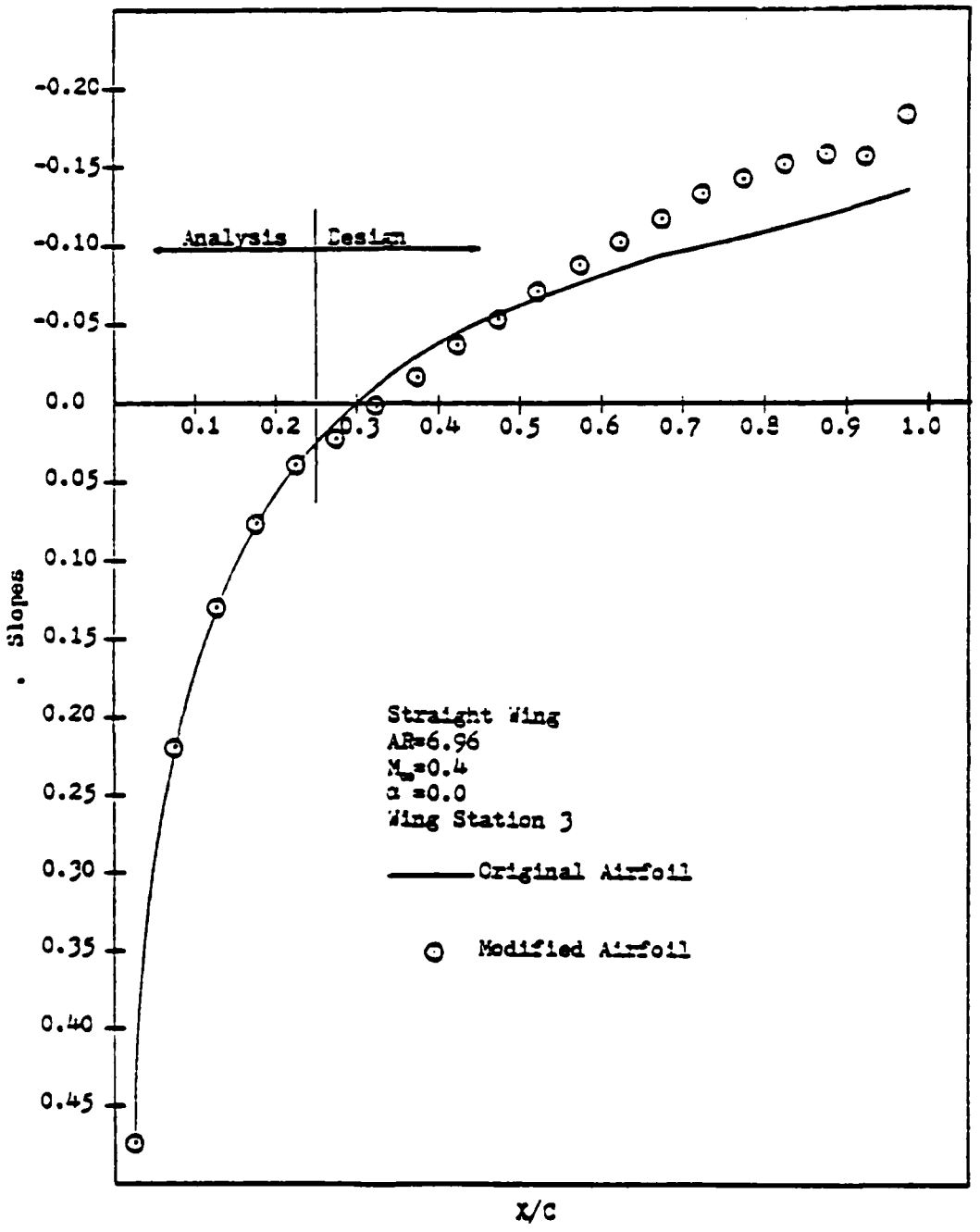


Figure 11. Airfoil slopes obtained from modified target pressures
Mach = 0.4, η = 0.1875

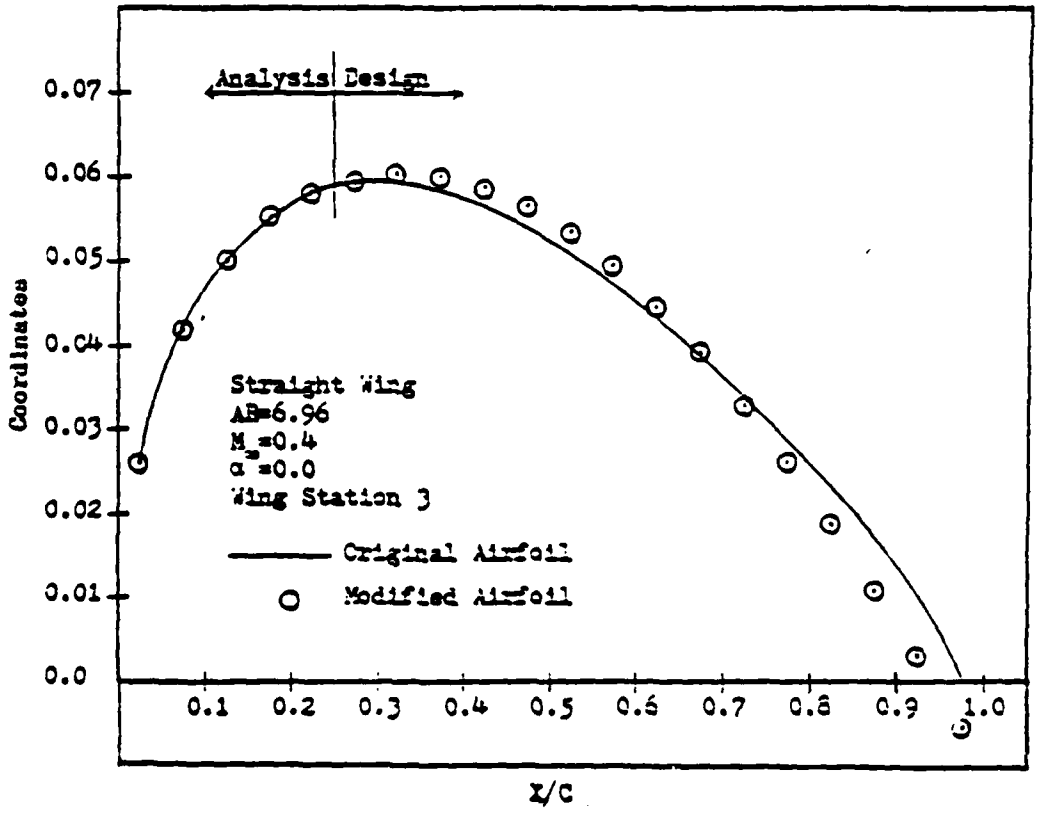


Figure 12. Airfoil ordinates obtained from modified target pressures Mach = 0.4, η = 0.1875

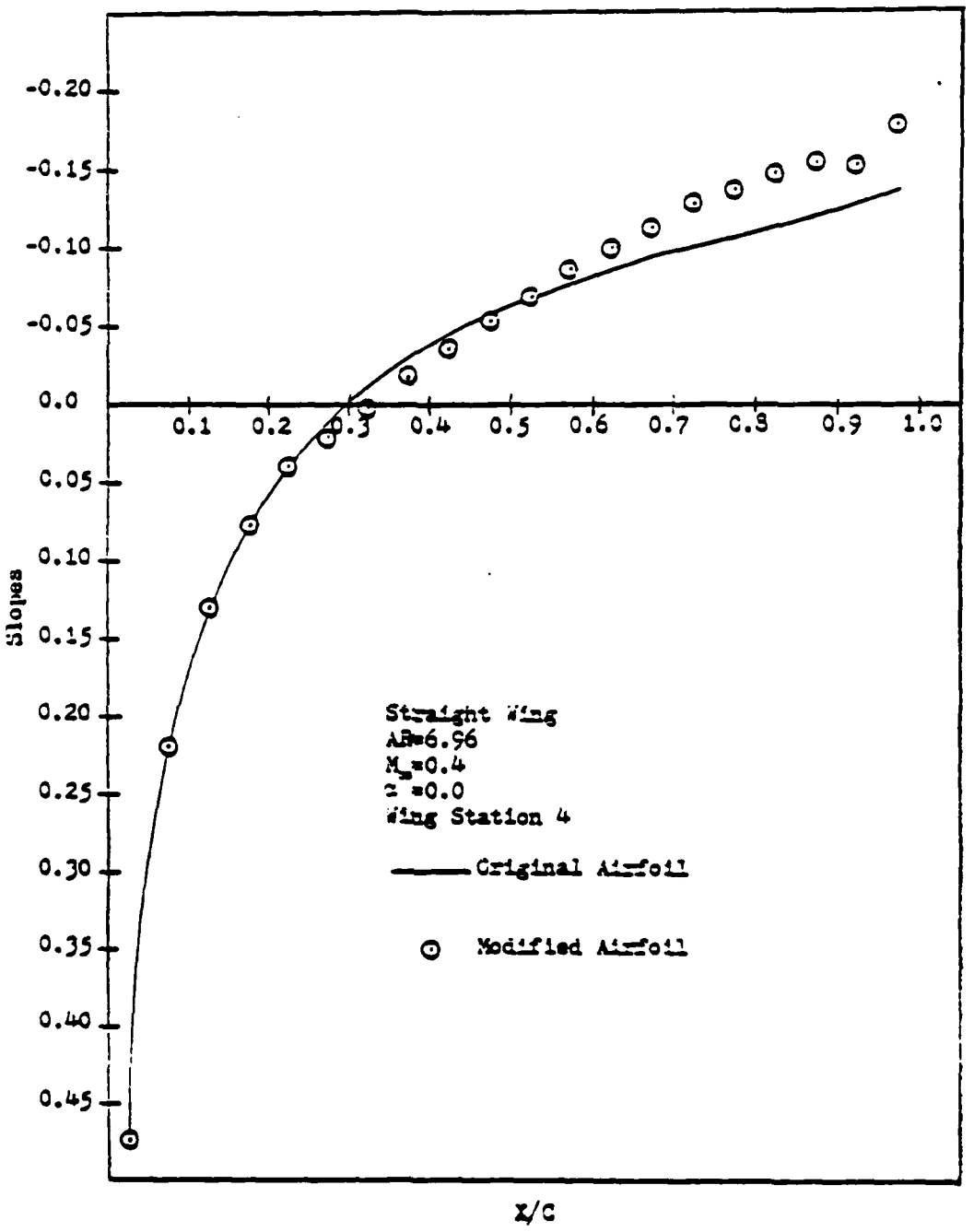


Figure 13. Airfoil slopes obtained from modified target pressures
Mach = 0.4, η = 0.3125

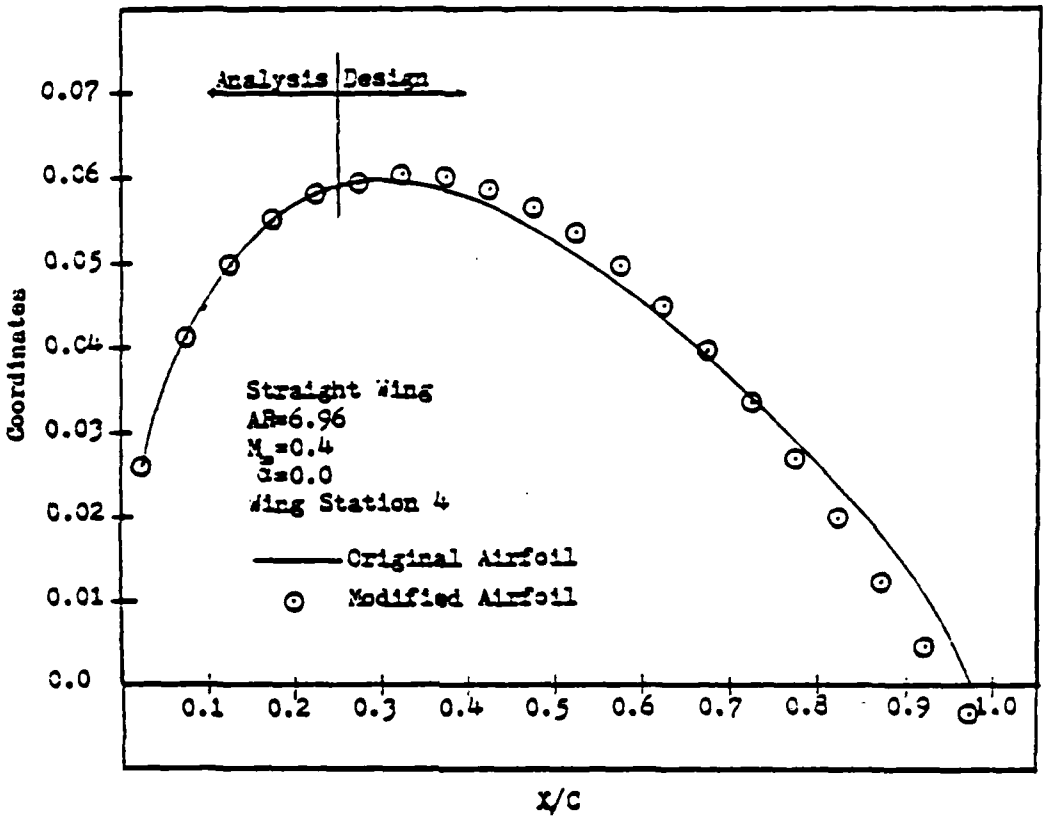


Figure 14. Airfoil ordinates obtained from modified target pressures Mach = 0.4, η = 0.3125

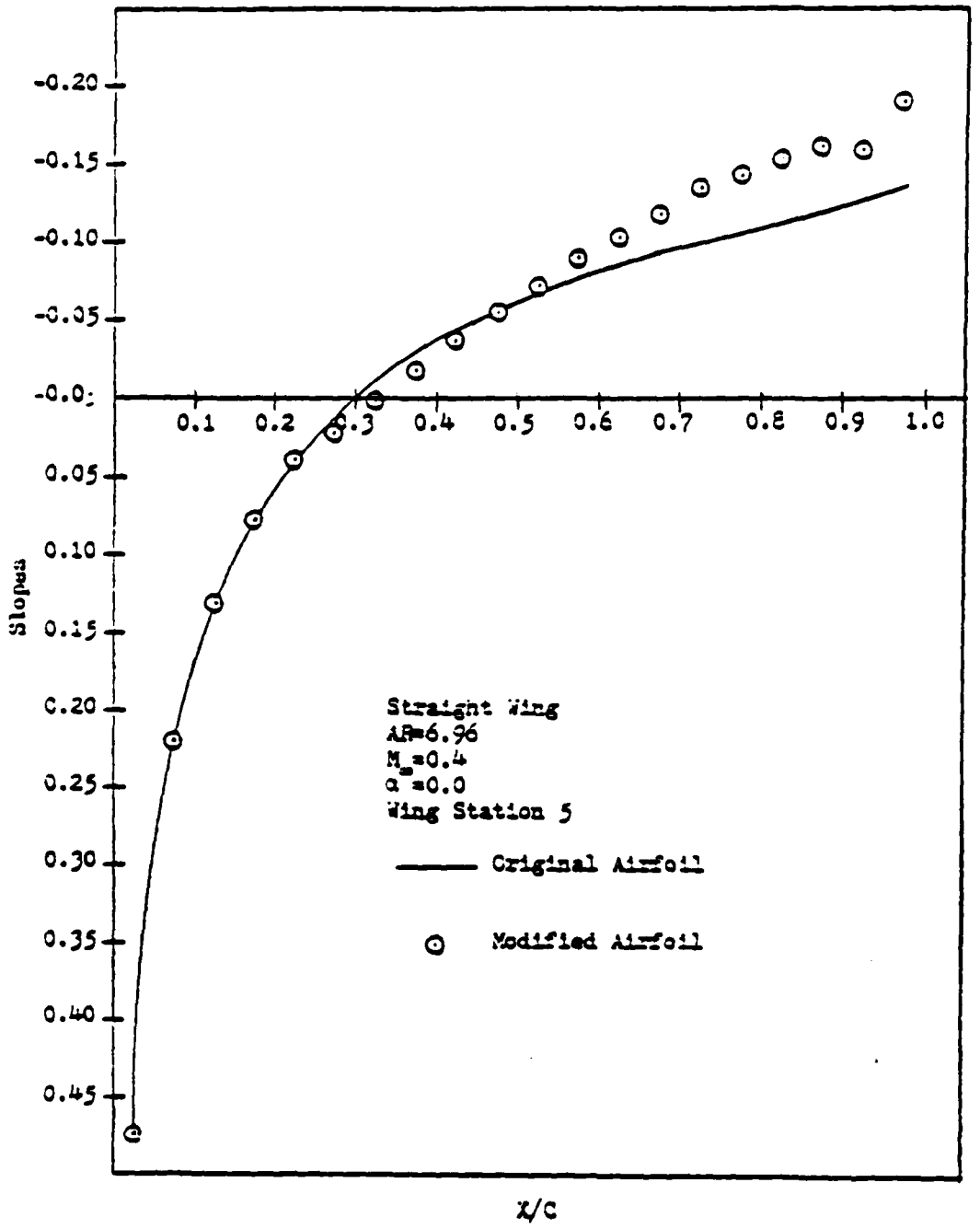


Figure 15. Airfoil slopes obtained from modified target pressures
Mach = 0.4, η = 0.4375

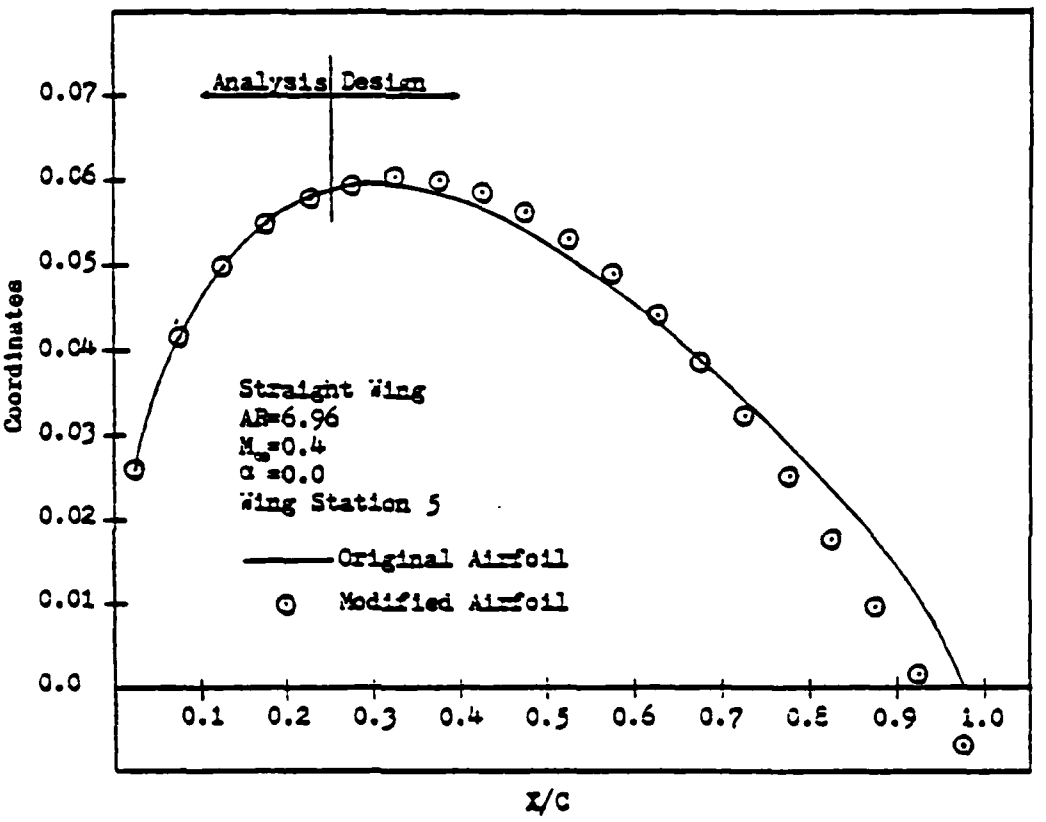


Figure 16. Airfoil ordinates obtained from modified target pressures
Mach = 0.4, η = 0.4375

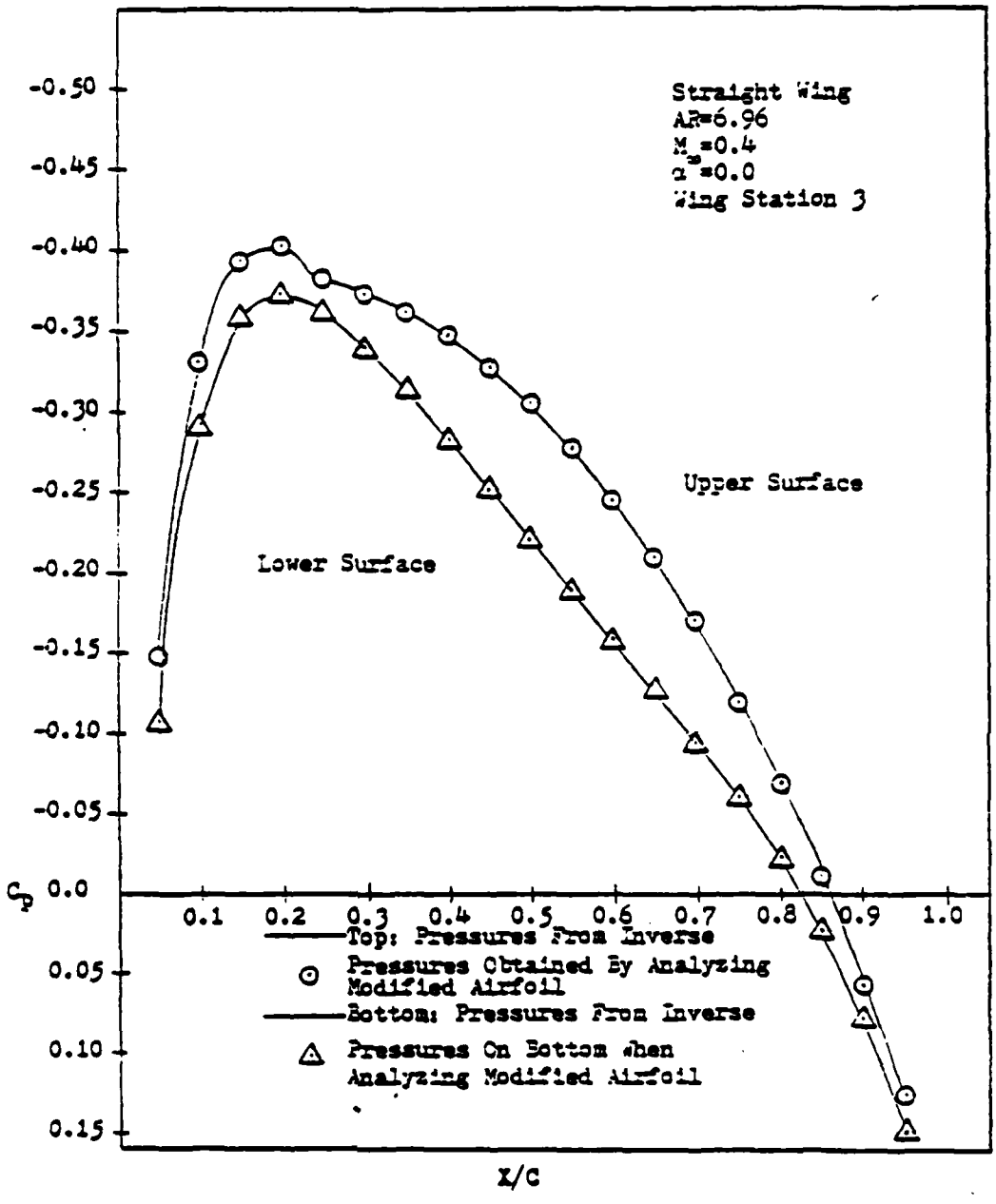


Figure 17. Correlation of inverse pressures and pressures from analysis of modified wing, Mach = 0.4, η = 0.1875

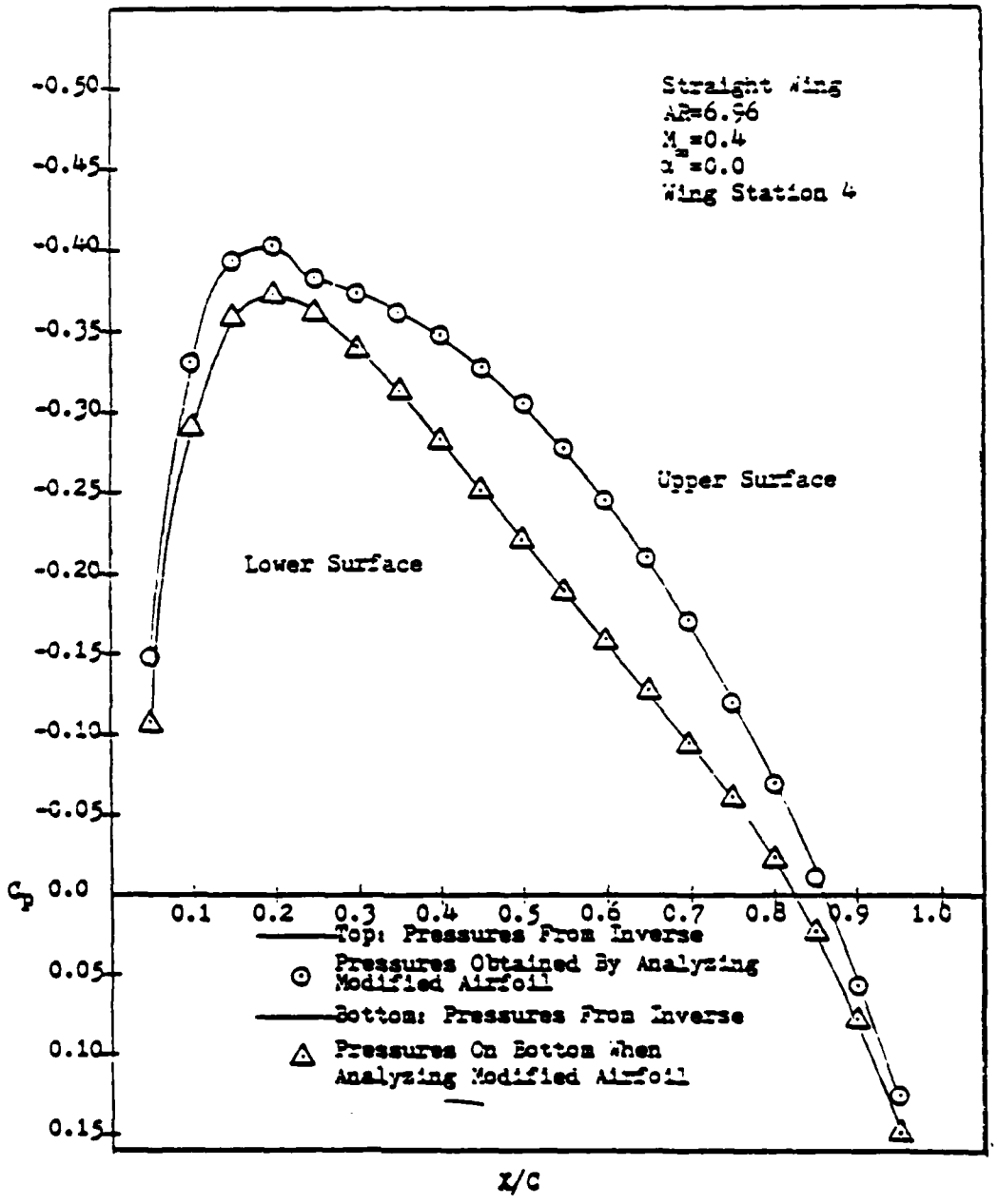


Figure 18. Correlation of inverse pressures and pressures from analysis of modified wing, Mach = 0.4, η = 0.3125

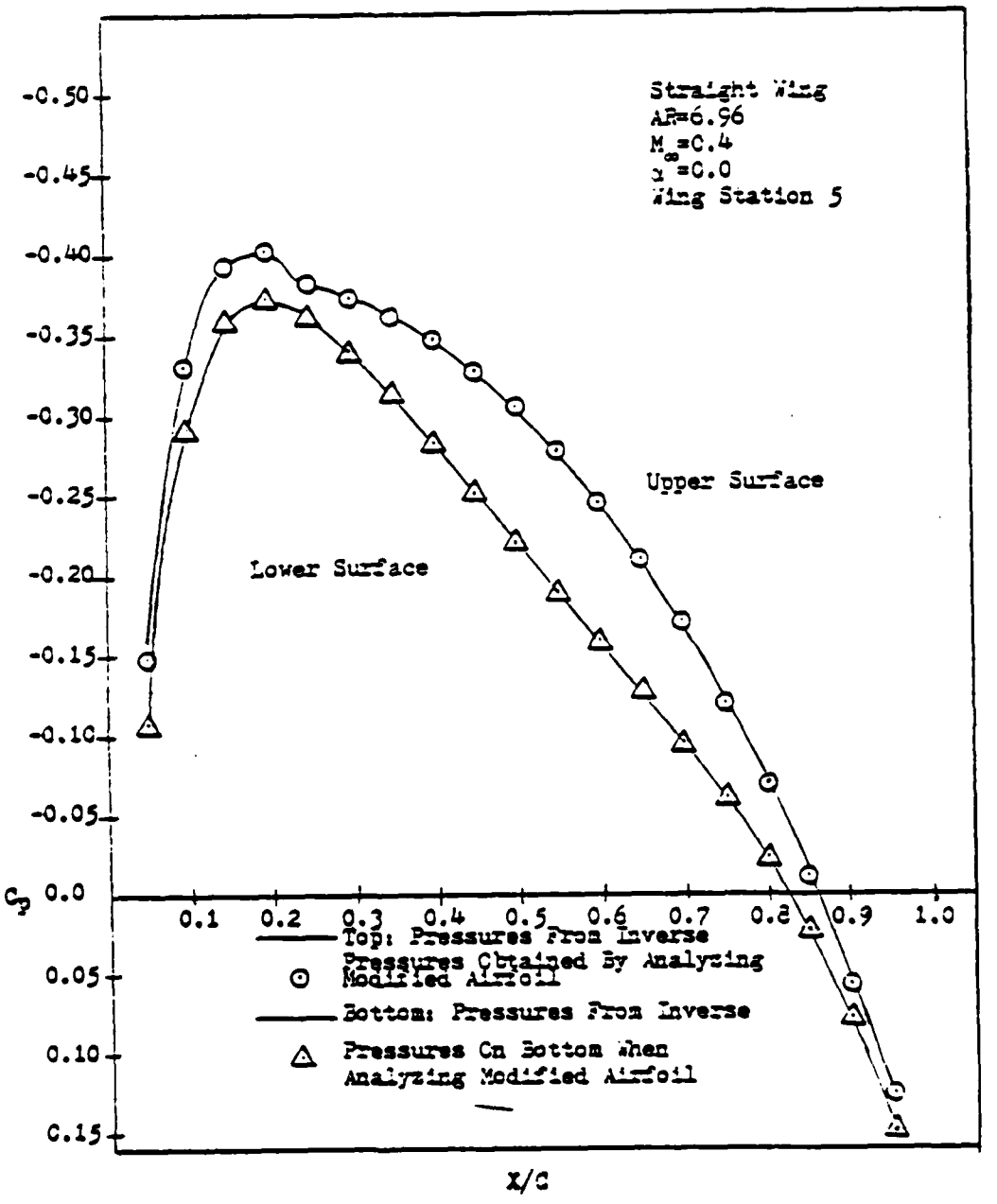


Figure 19. Correlation of inverse pressures and pressures from analysis of modified wing, Mach = 0.4, η = 0.4375

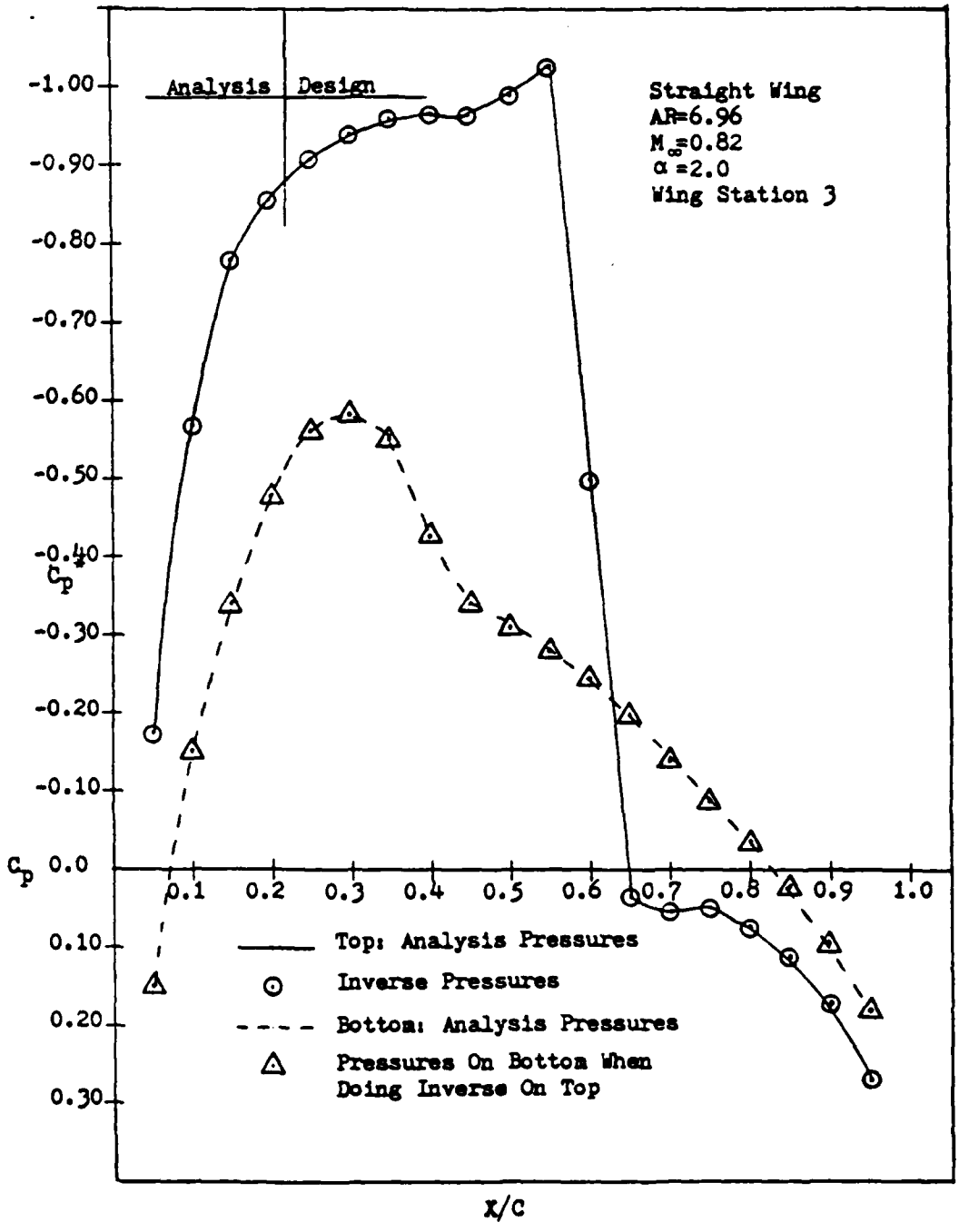


Figure 20. Correlation of inverse and analysis target pressures, Mach = 0.82, α = 2.0, η = 0.1875

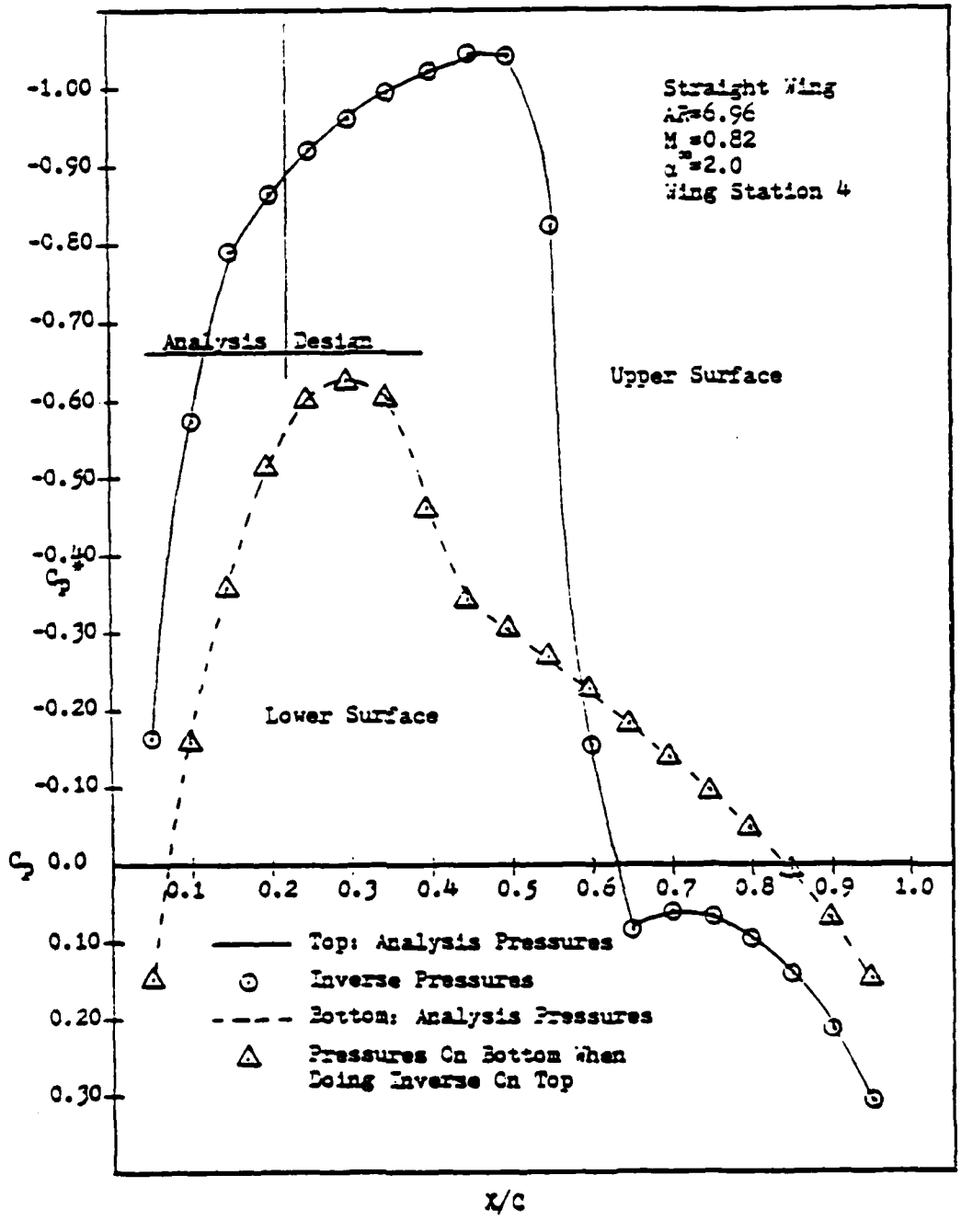


Figure 21. Correlation of inverse and analysis target pressures, Mach = 0.82, α = 2.0, η = 0.3125

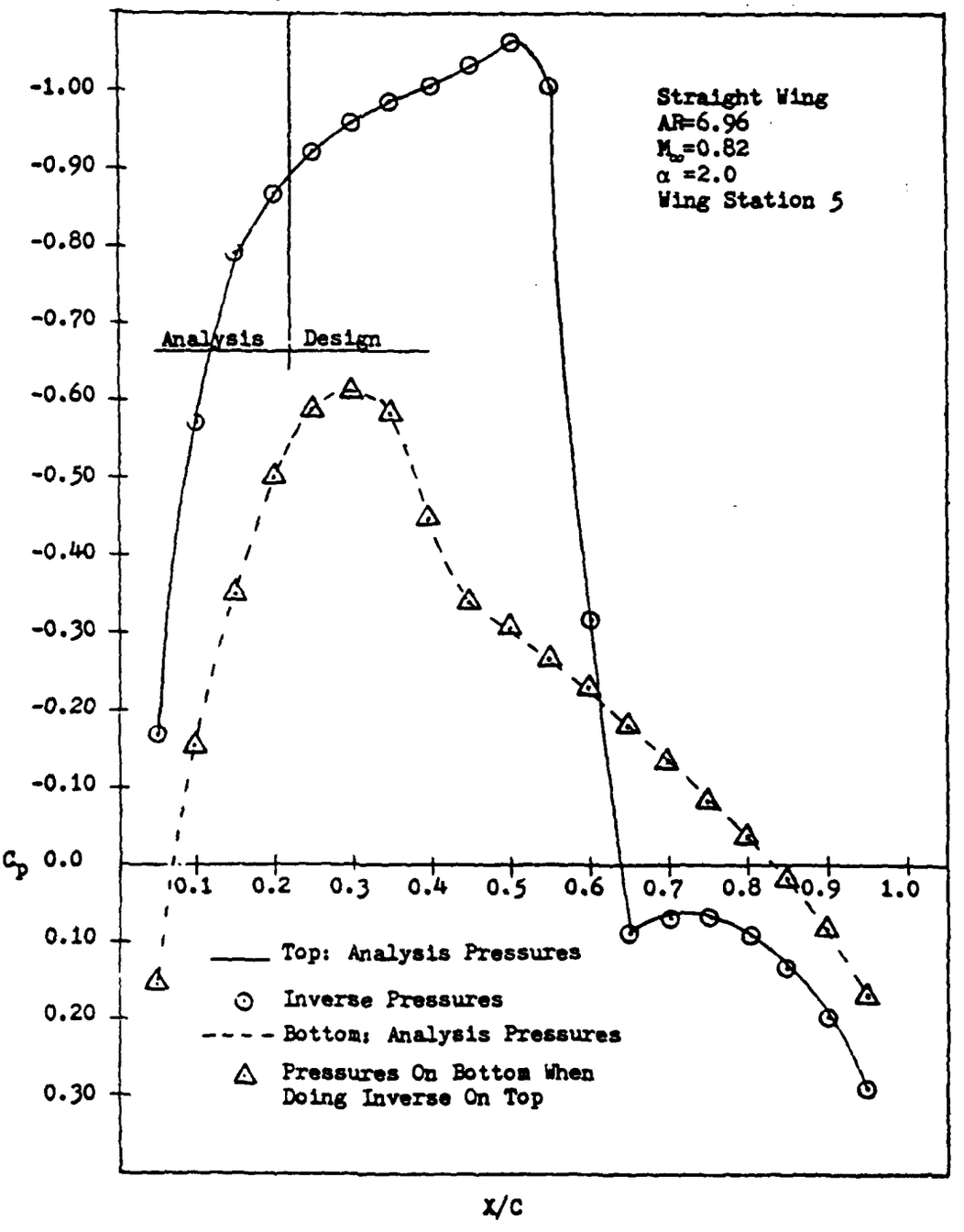


Figure 22. Correlation of inverse and analysis target pressures, Mach = 0.82, α = 2.0, η = 0.4375

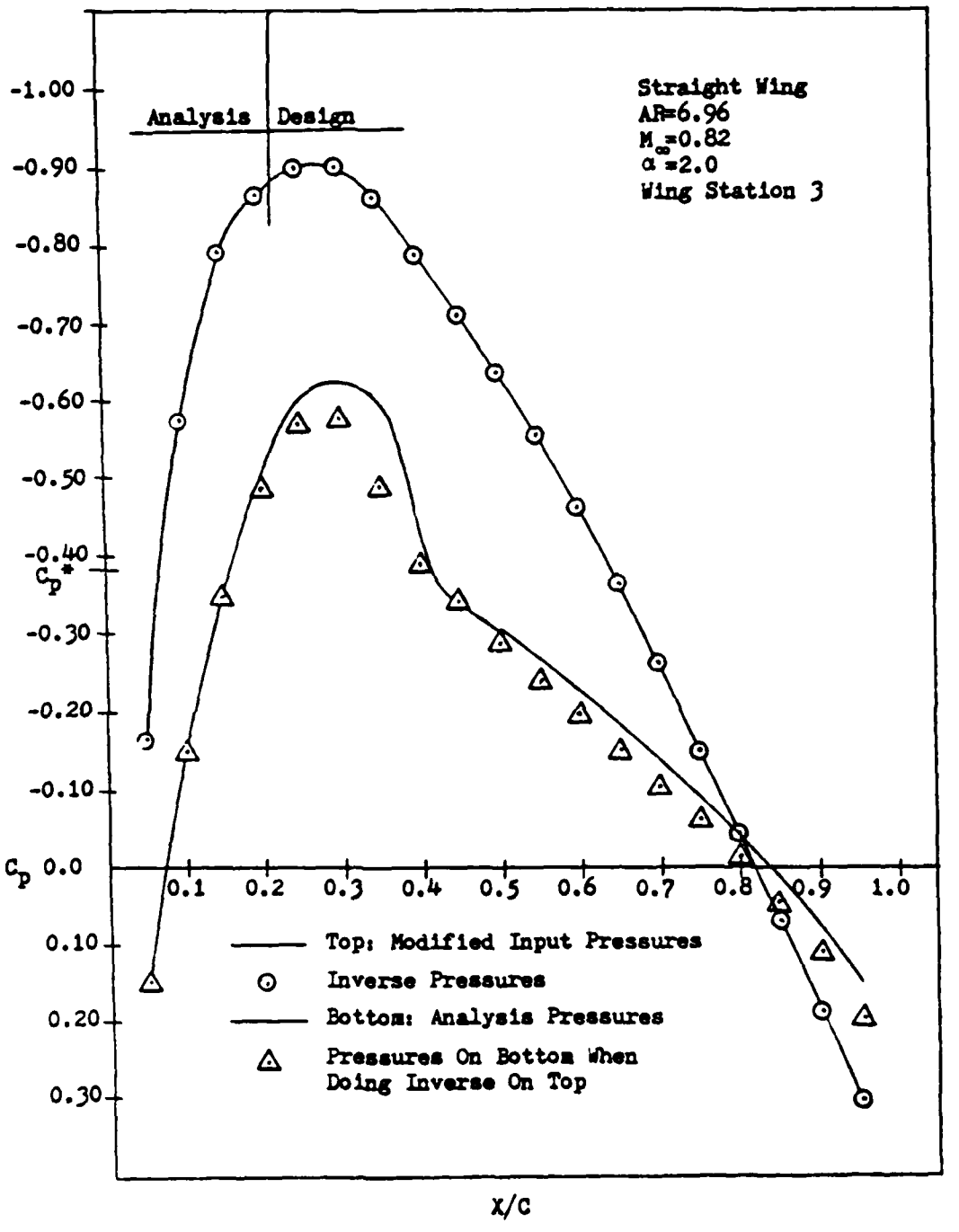


Figure 23. Correlation of inverse and modified target pressures, Mach = 0.82, α = 2.0, η = 0.1875

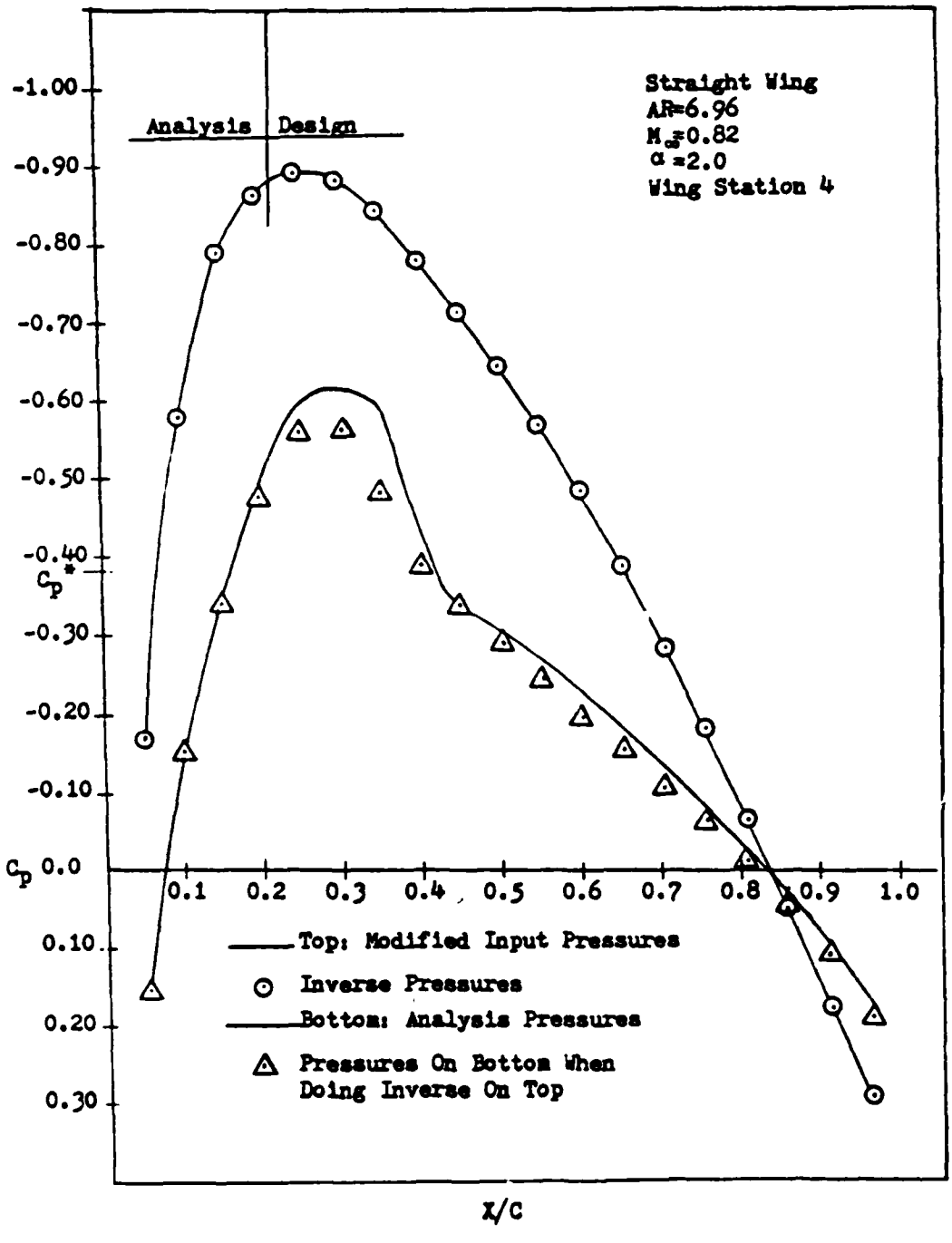


Figure 24. Correlation of inverse and modified target pressures, Mach = 0.82, α = 2.0, η = 0.3125

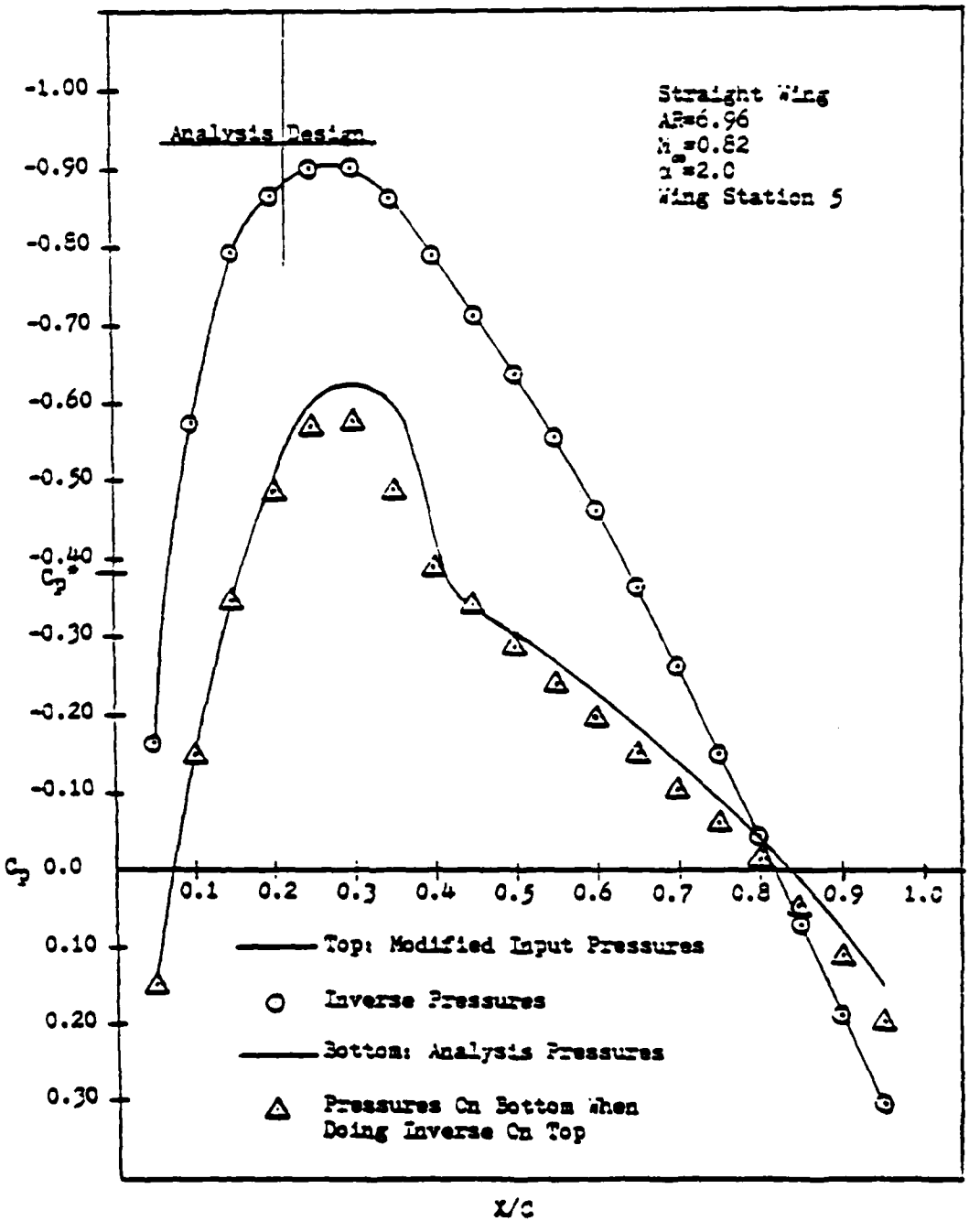


Figure 25. Correlation of inverse and modified target pressures, Mach = 0.82, α = 2.0, η = 0.4375

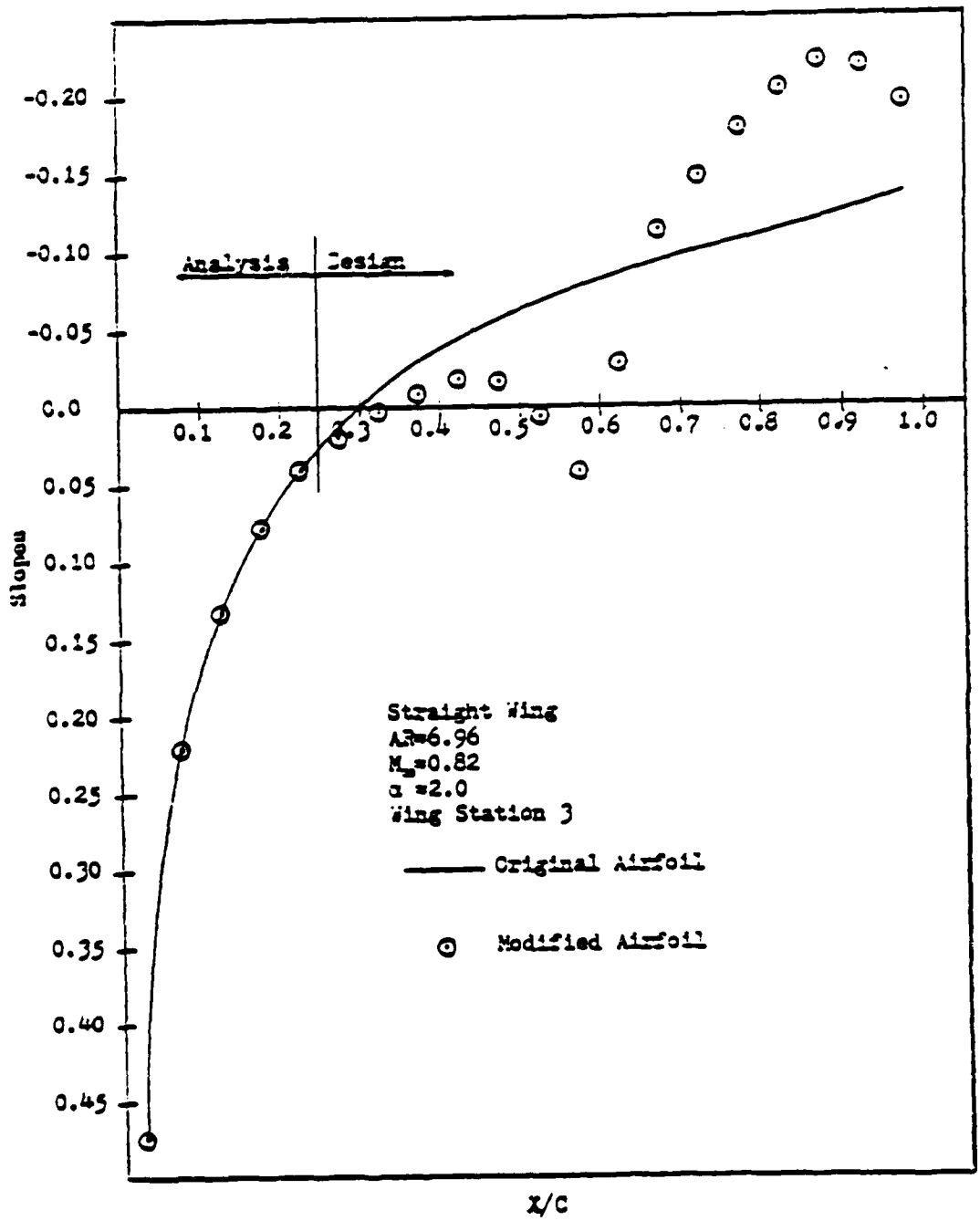


Figure 26. Airfoil slopes from modified target pressures, Mach = 0.82, α = 2.0, η = 0.1875

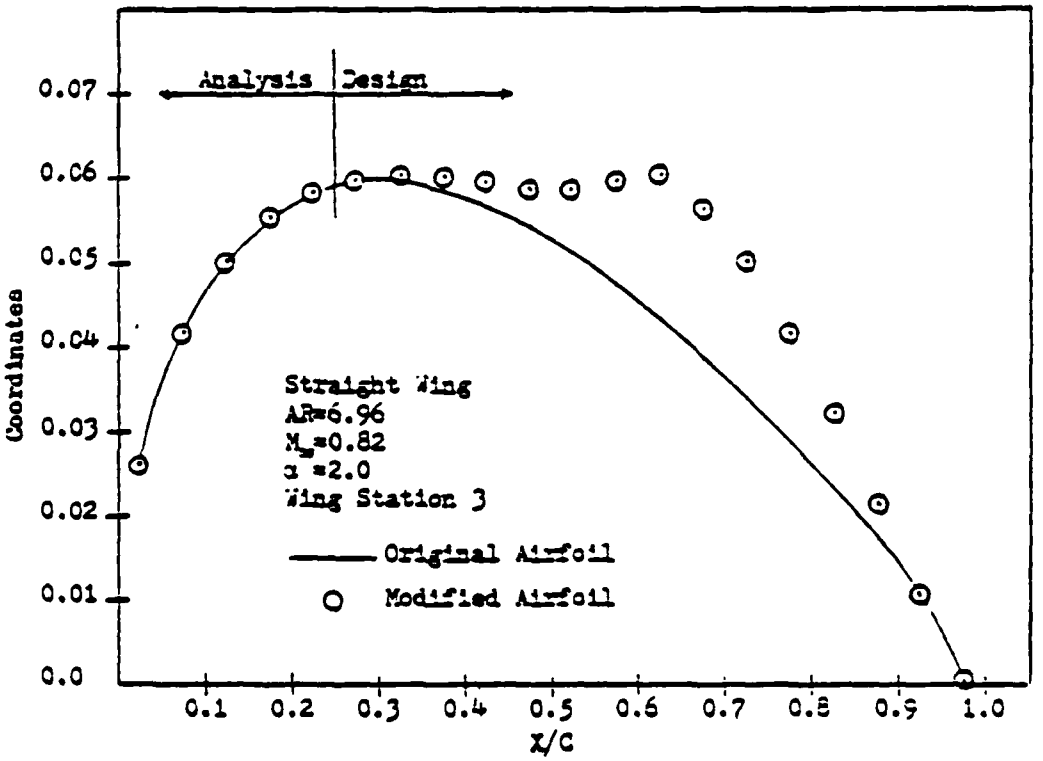


Figure 27. Airfoil ordinates from modified target pressures, Mach = 0.82, α = 2.0, η = 0.1875

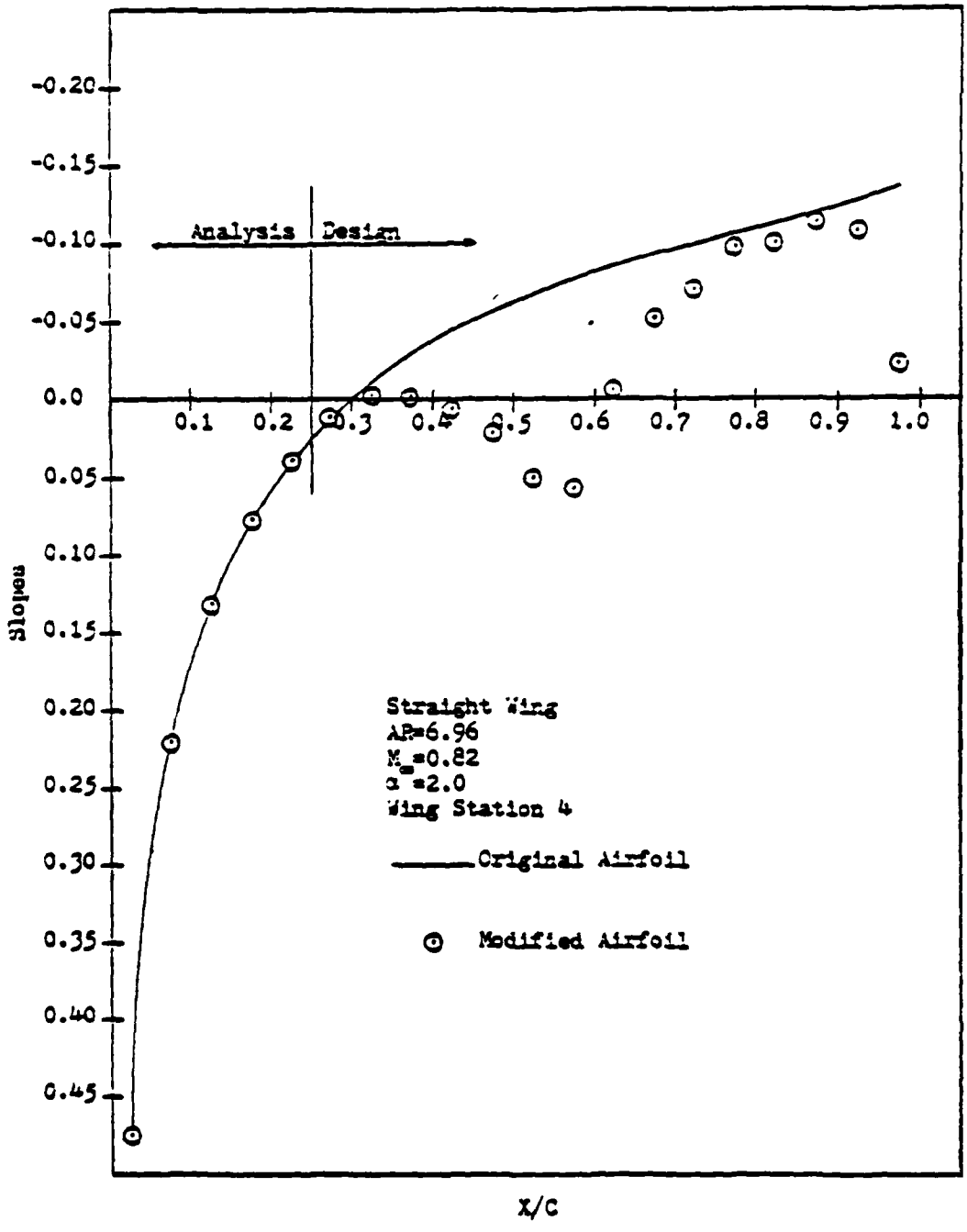


Figure 28. Airfoil slopes from modified target pressures, Mach = 0.82, α = 2.0, η = 0.3125

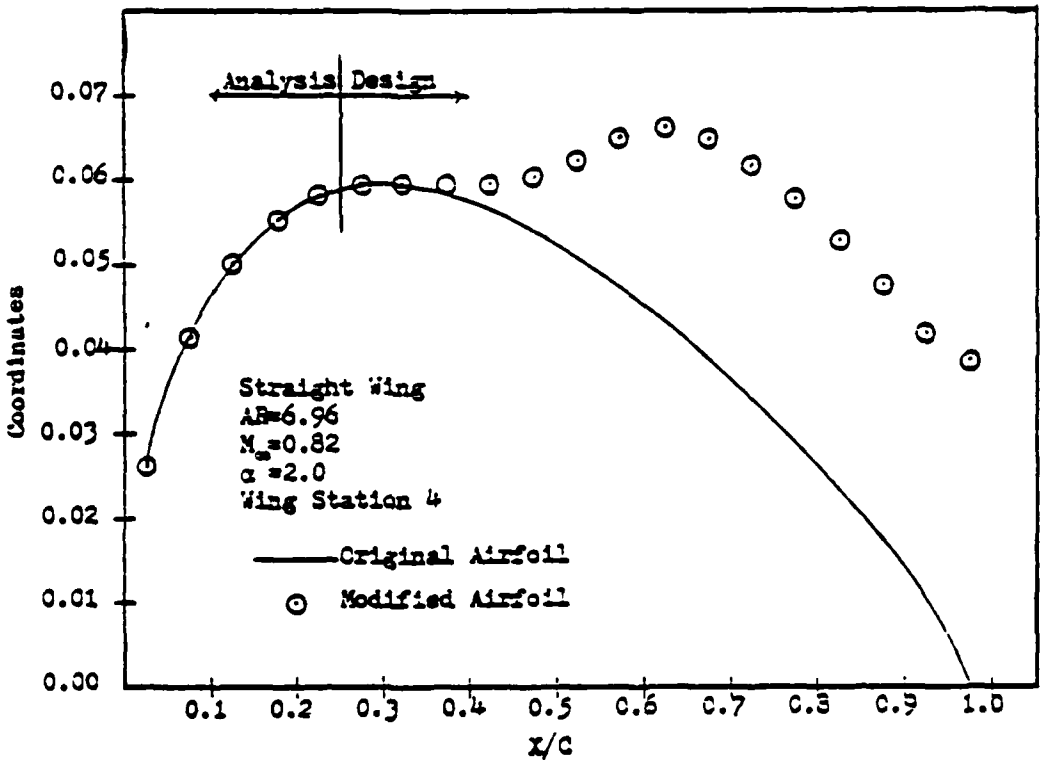


Figure 29. Airfoil ordinates from modified target pressures, Mach = 0.82, α = 2.0, η = 0.3125

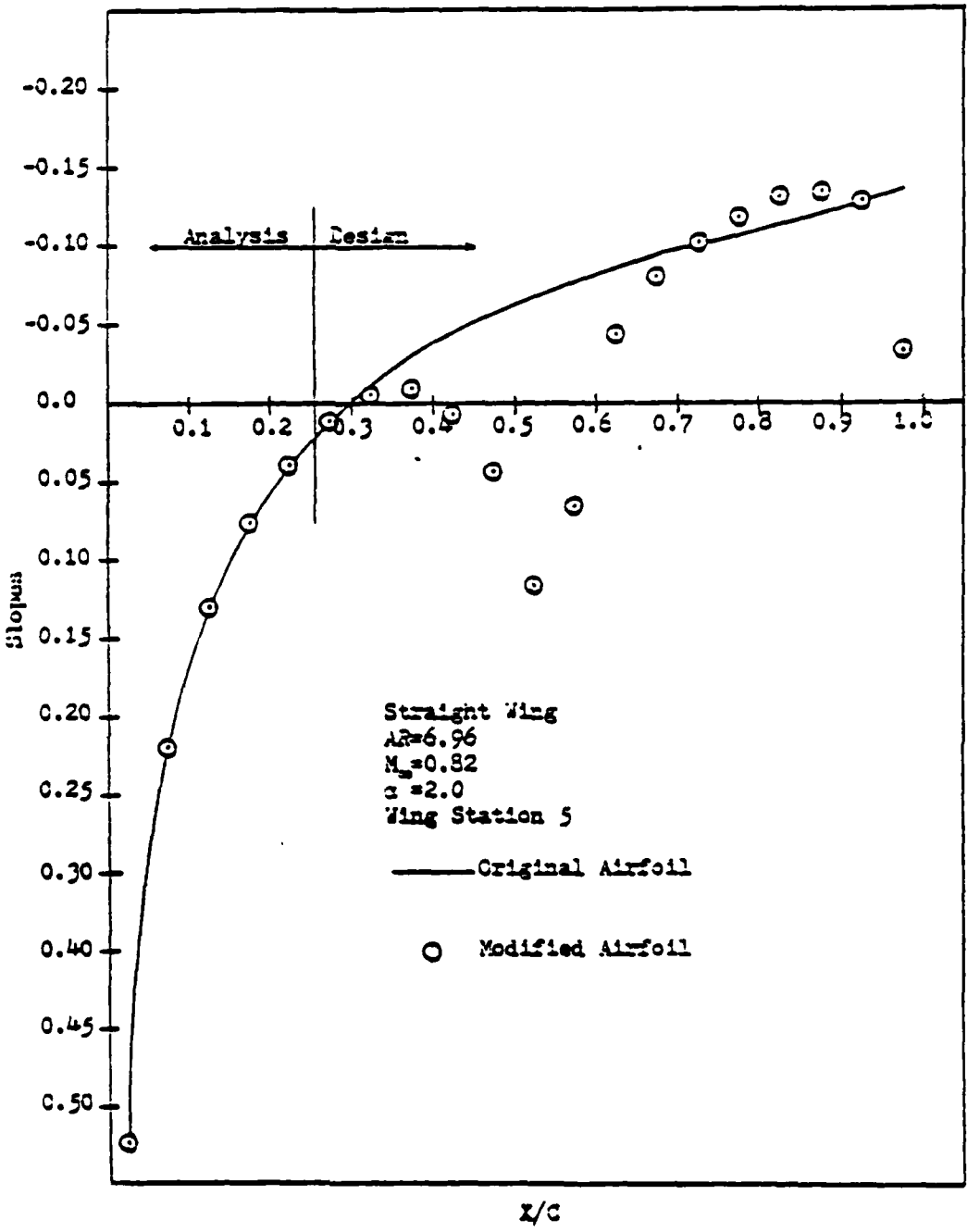


Figure 30. Airfoil slopes from modified target pressures, Mach = 0.82, α = 2.0, η = 0.4375

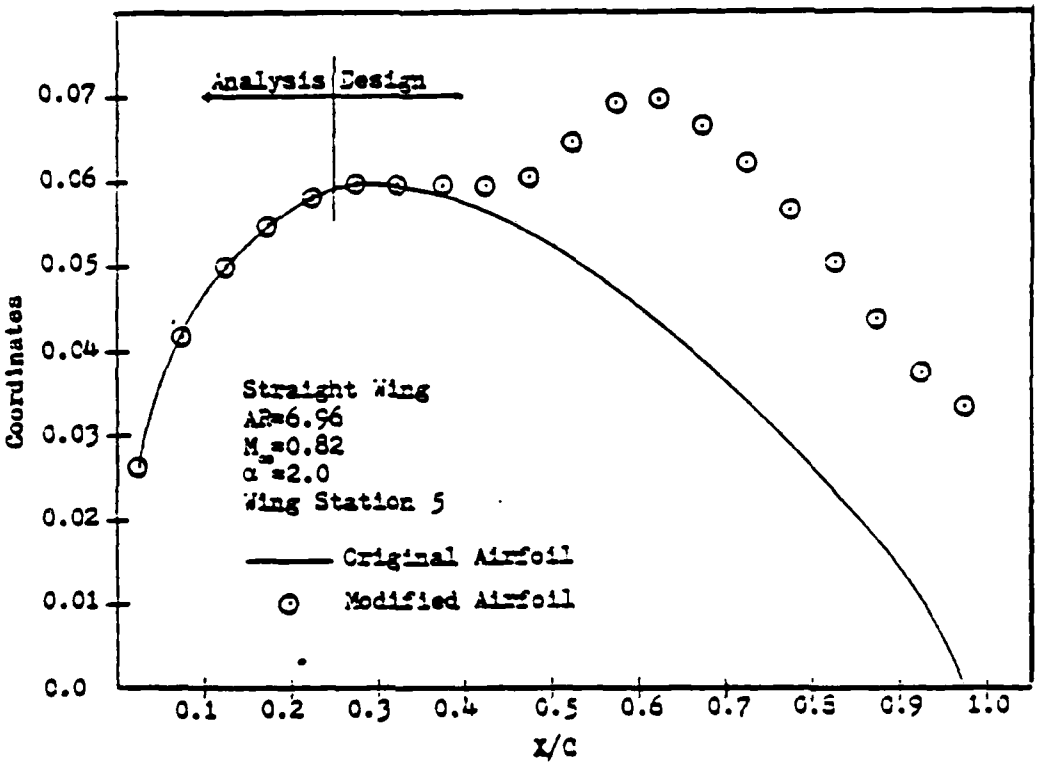


Figure 31. Airfoil ordinates from modified target pressures, Mach = 0.82, α = 2.0, η = 0.4375

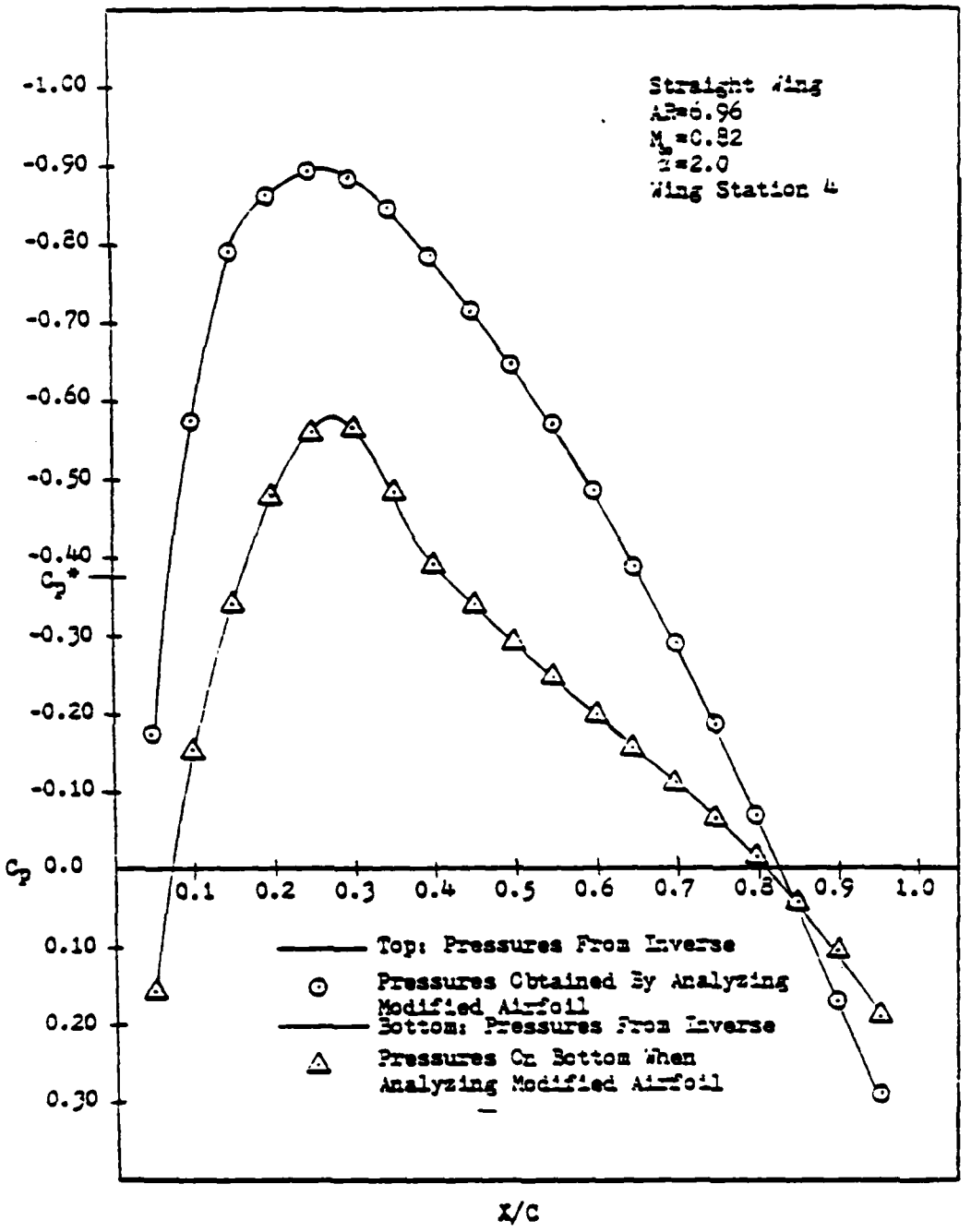


Figure 32. Comparison of inverse pressures with pressures from analysis of modified wing, $\eta = 0.1875$

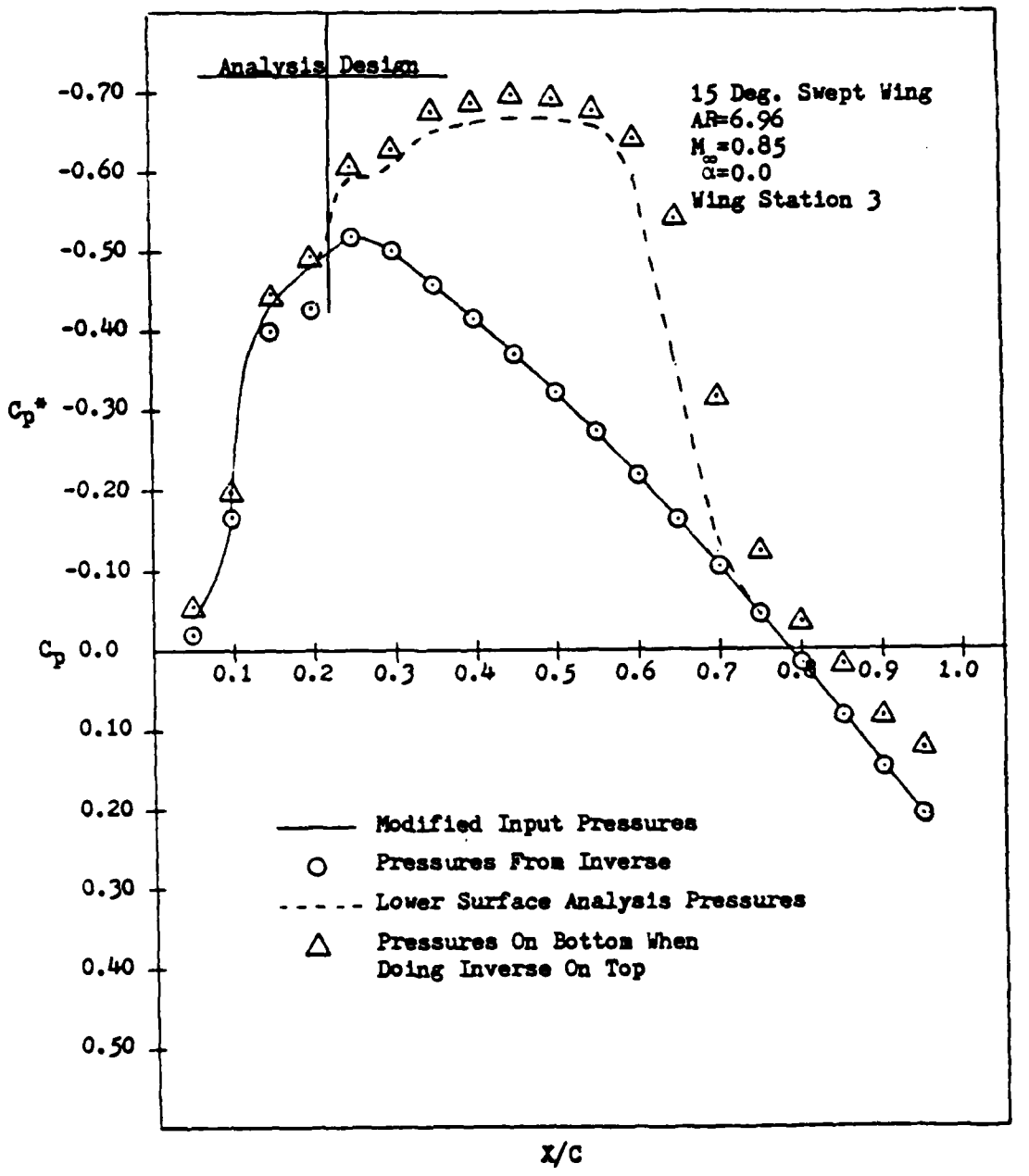


Figure 33. Correlation of modified target pressures and inverse pressures for a swept wing, Mach = 0.85, λ = 2.0, η = 0.1875

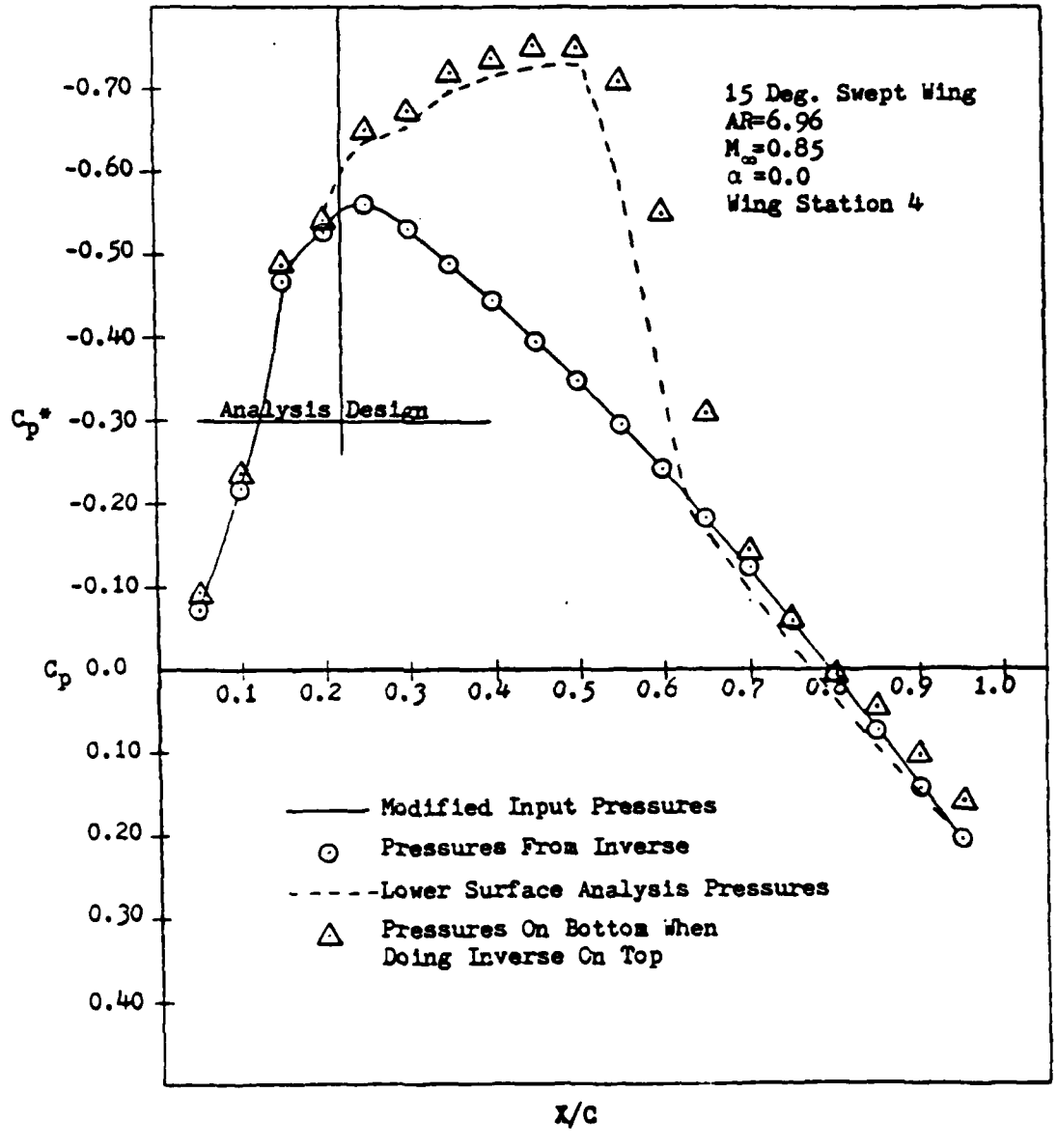


Figure 34. Correlation of modified target pressures and inverse pressures for a swept wing, Mach = 0.85, α = 2.0, η = 0.3125

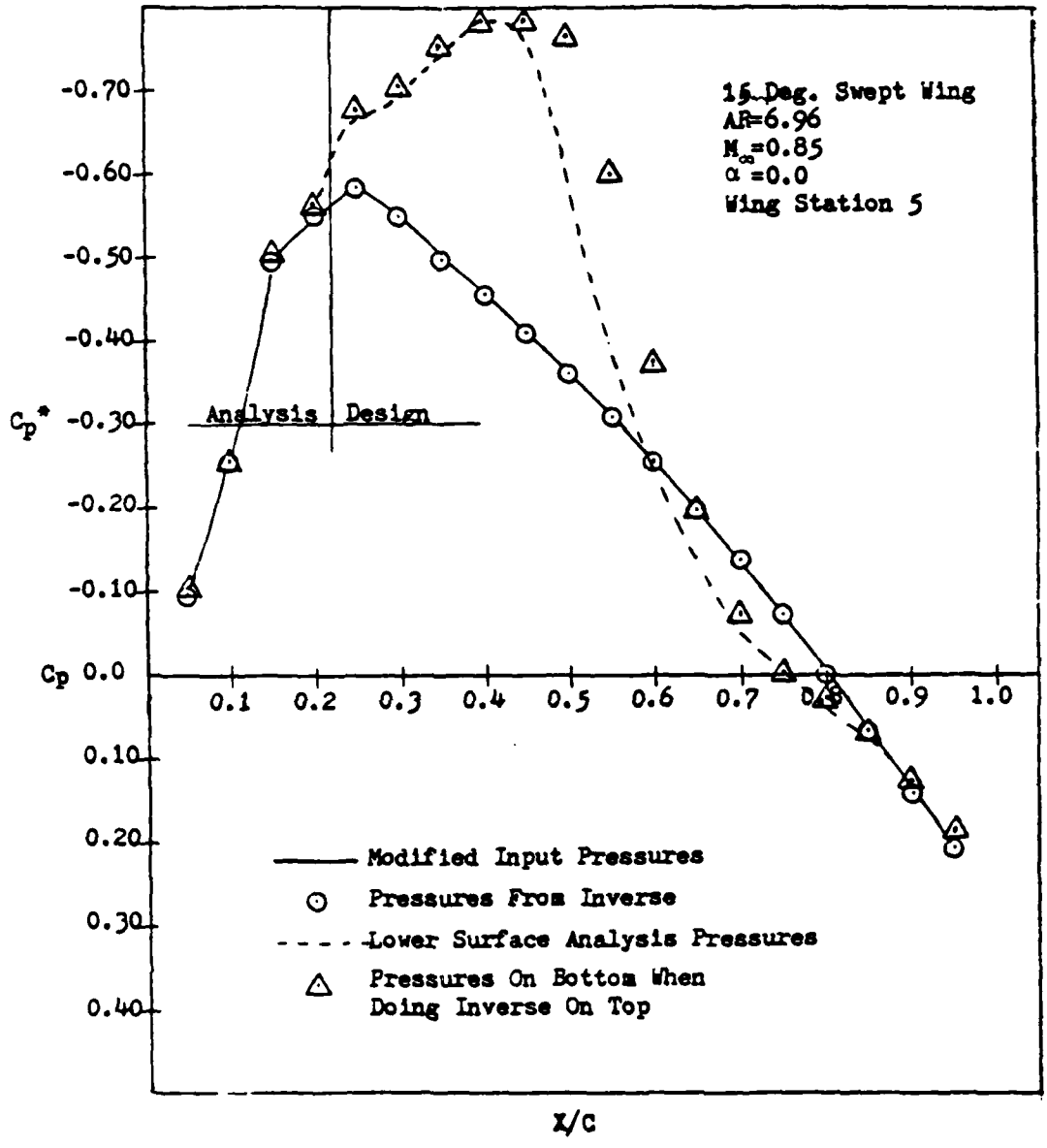


Figure 35. Correlation of modified target pressures and inverse pressures for a swept wing, Mach = 0.85, α = 2.0, η = 0.4375

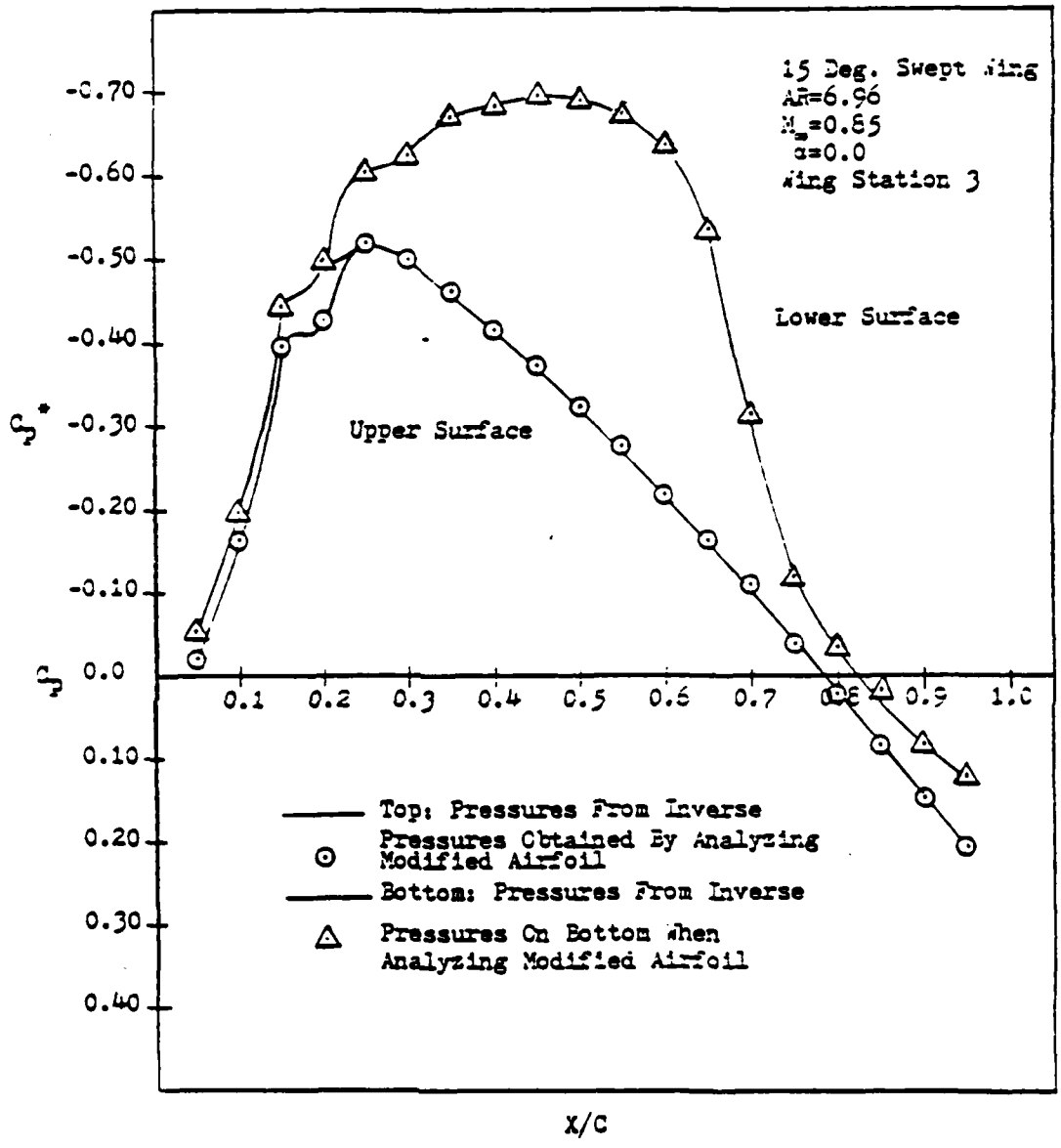


Figure 36. Comparison of inverse pressures with pressures from analysis of modified swept wing, Mach = 0.85, α = 2.0, η = 0.1875

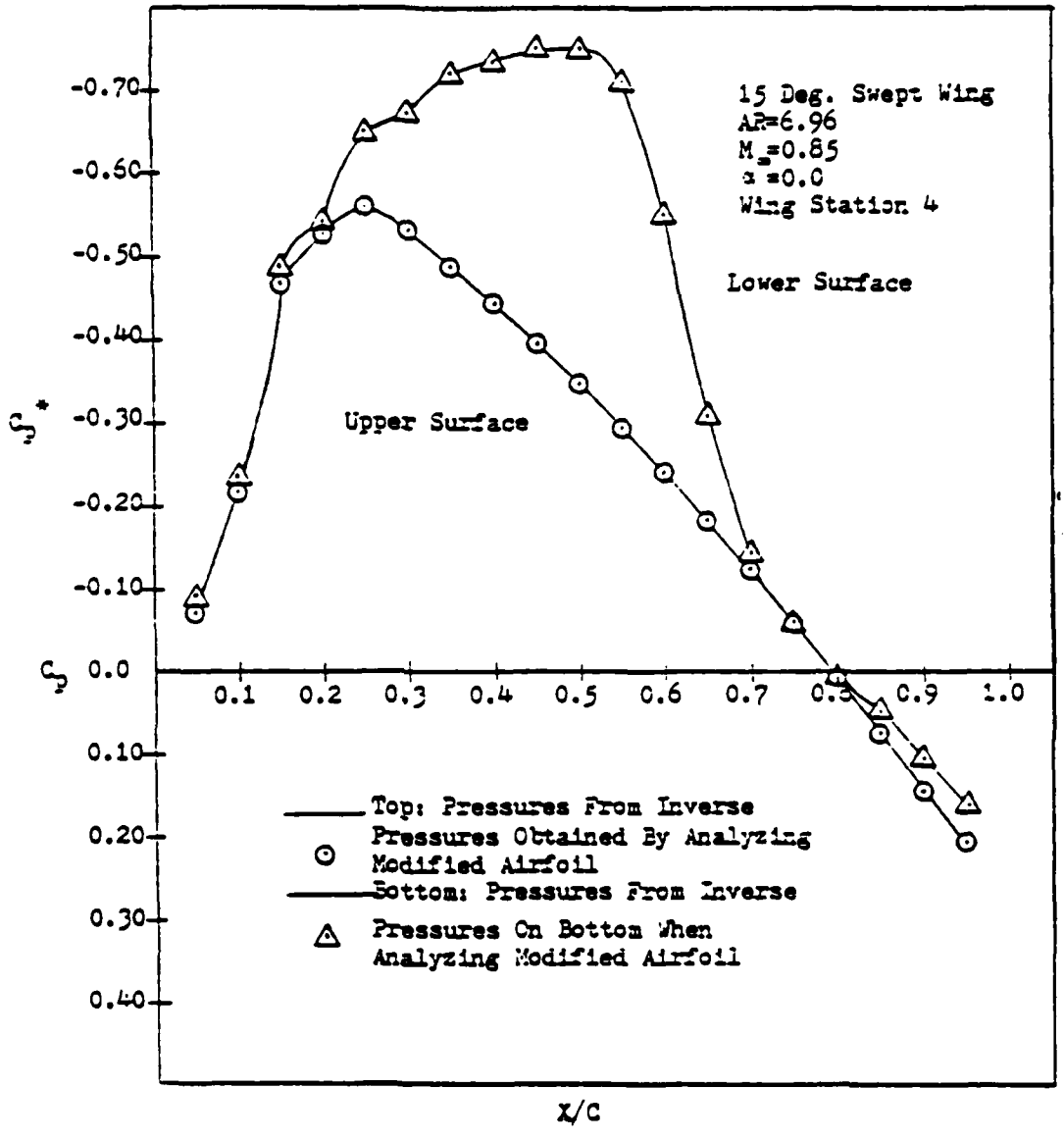


Figure 37. Comparison of inverse pressures with pressures from analysis of modified swept wing, Mach = 0.85, α = 2.0, η = 0.3125

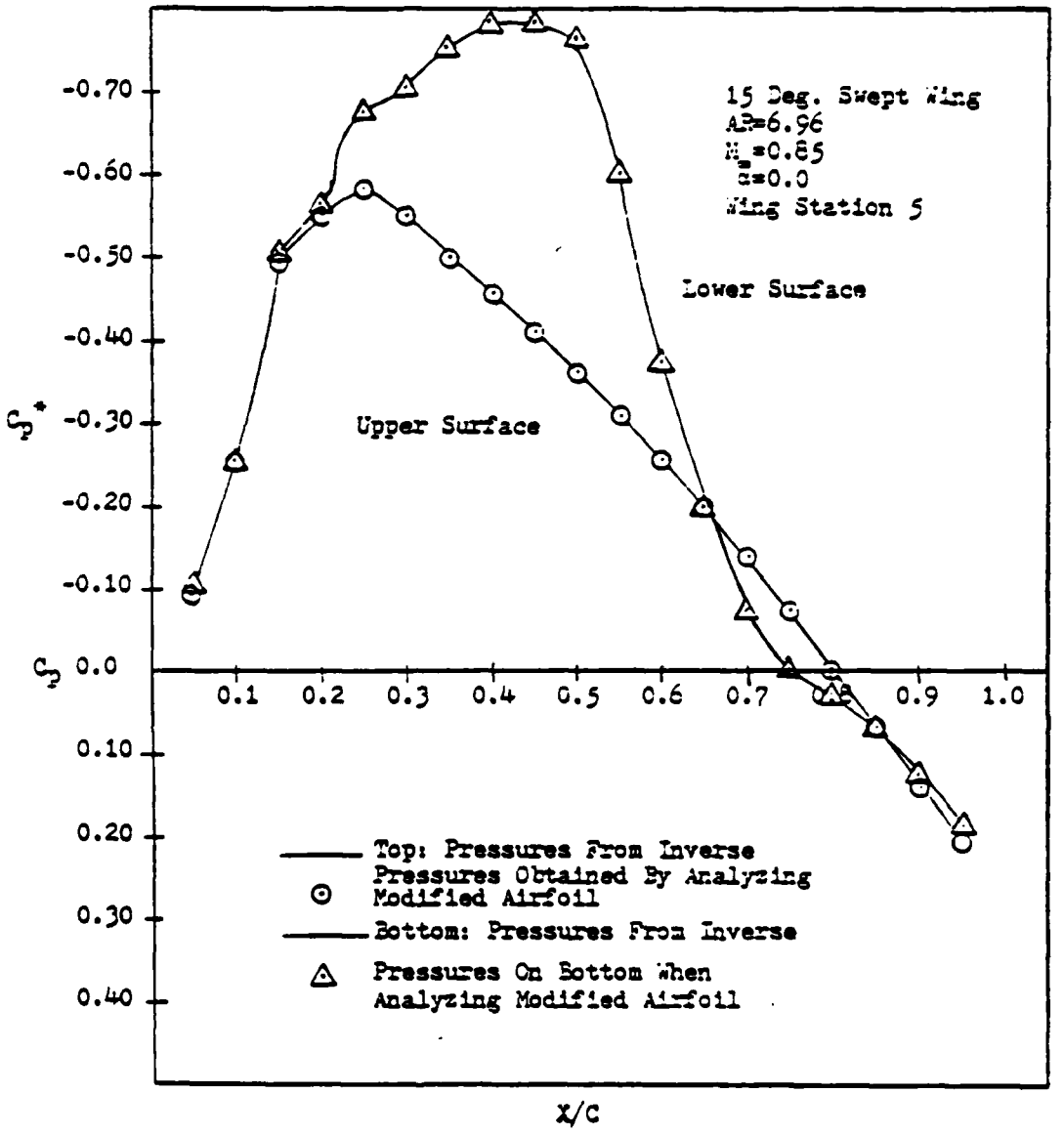


Figure 38. Comparison of inverse pressures with pressures from analysis of modified swept wing, Mach = 0.85, α = 2.0, η = 0.4375

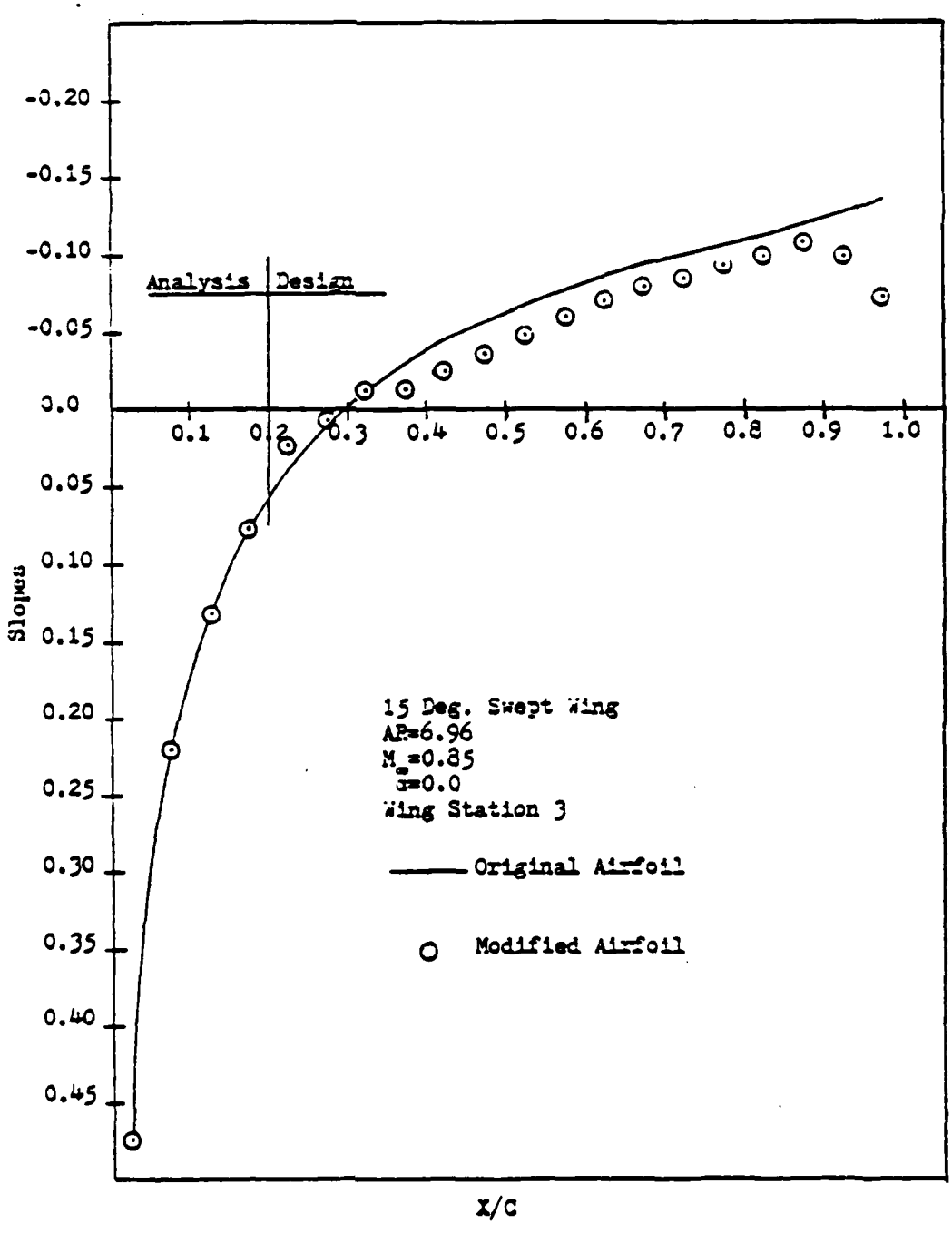


Figure 39. Modified swept wing airfoil slopes, $\eta = 0.1875$

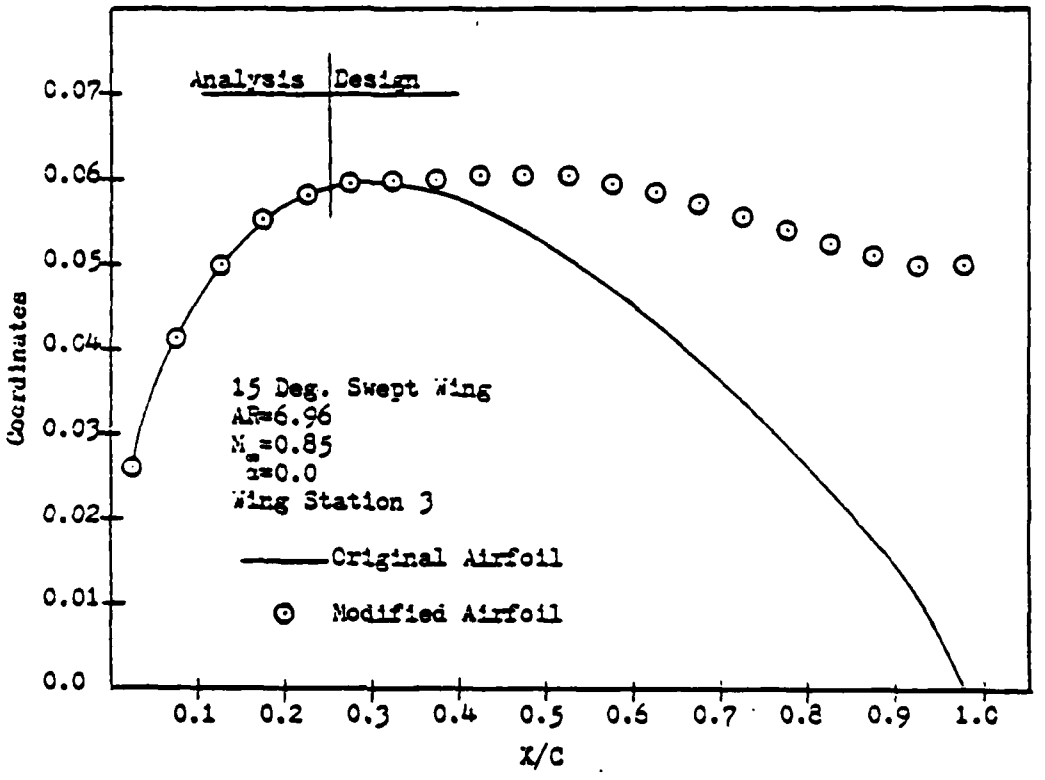


Figure 40. Modified swept wing airfoil ordinates, η = 0.1875

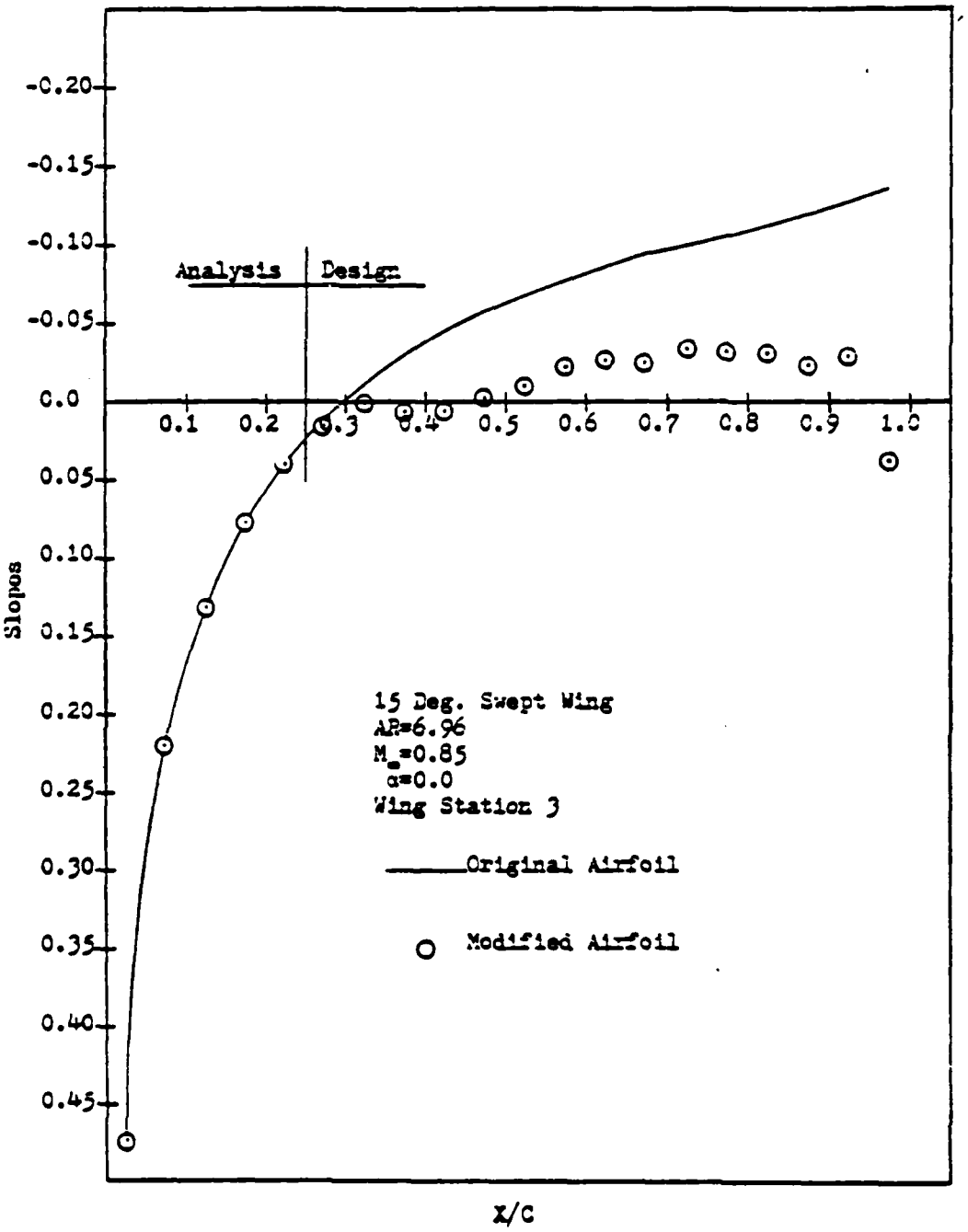


Figure 41. Modified swept wing airfoil slopes, $\eta = 0.3125$

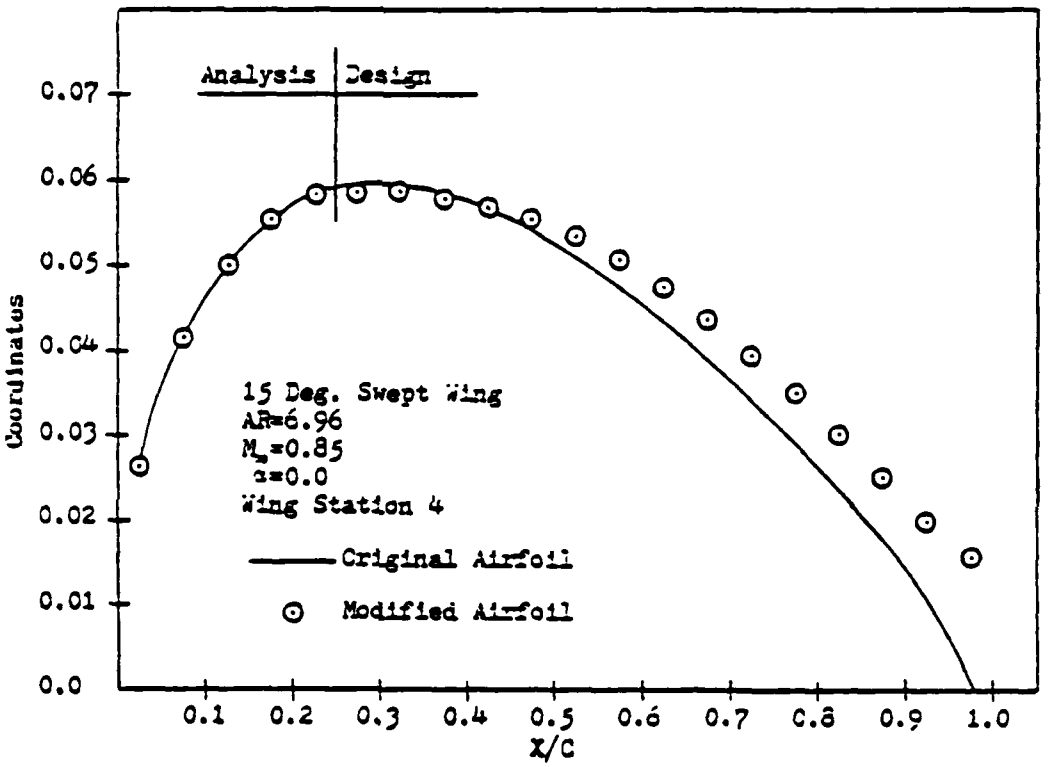


Figure 42. Modified swept wing airfoil ordinates, η = 0.3125

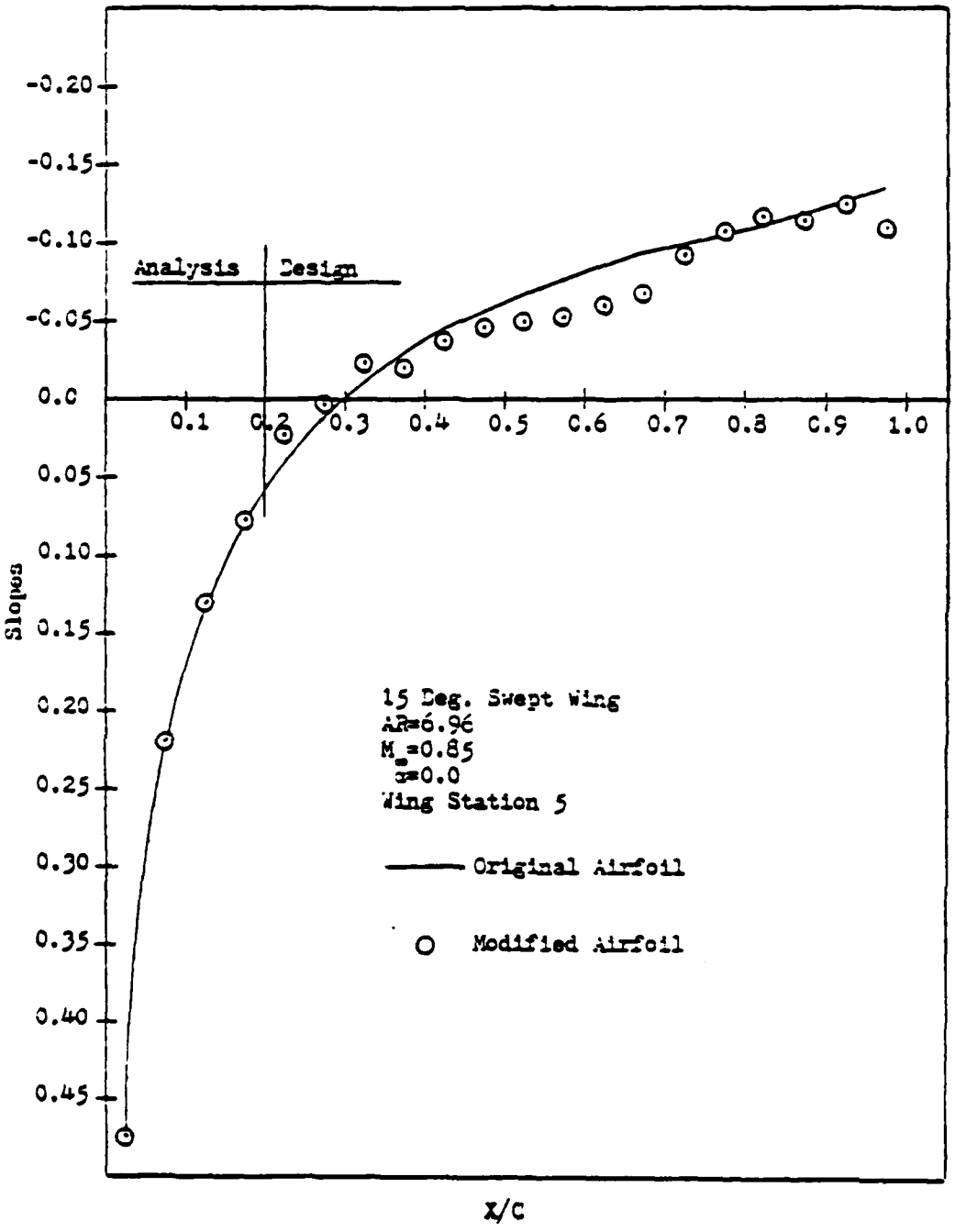


Figure 43. Modified swept wing airfoil slopes, $\eta = 0.4375$

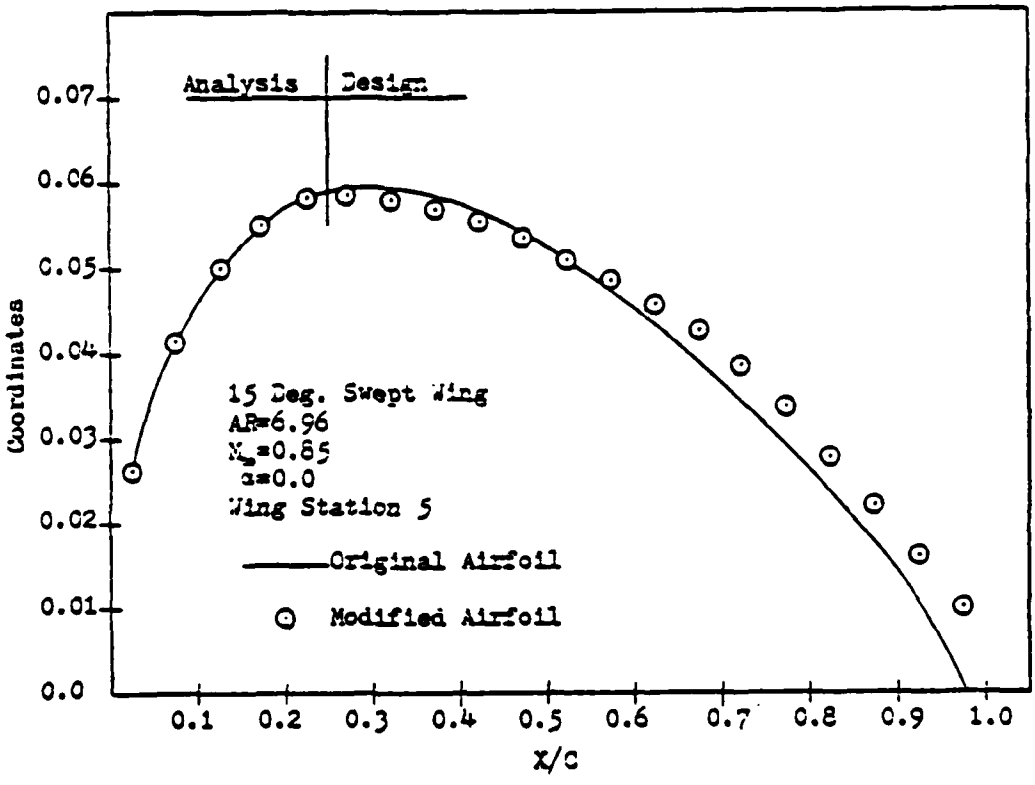


Figure 44. Modified swept wing airfoil ordinates, η = 0.4375

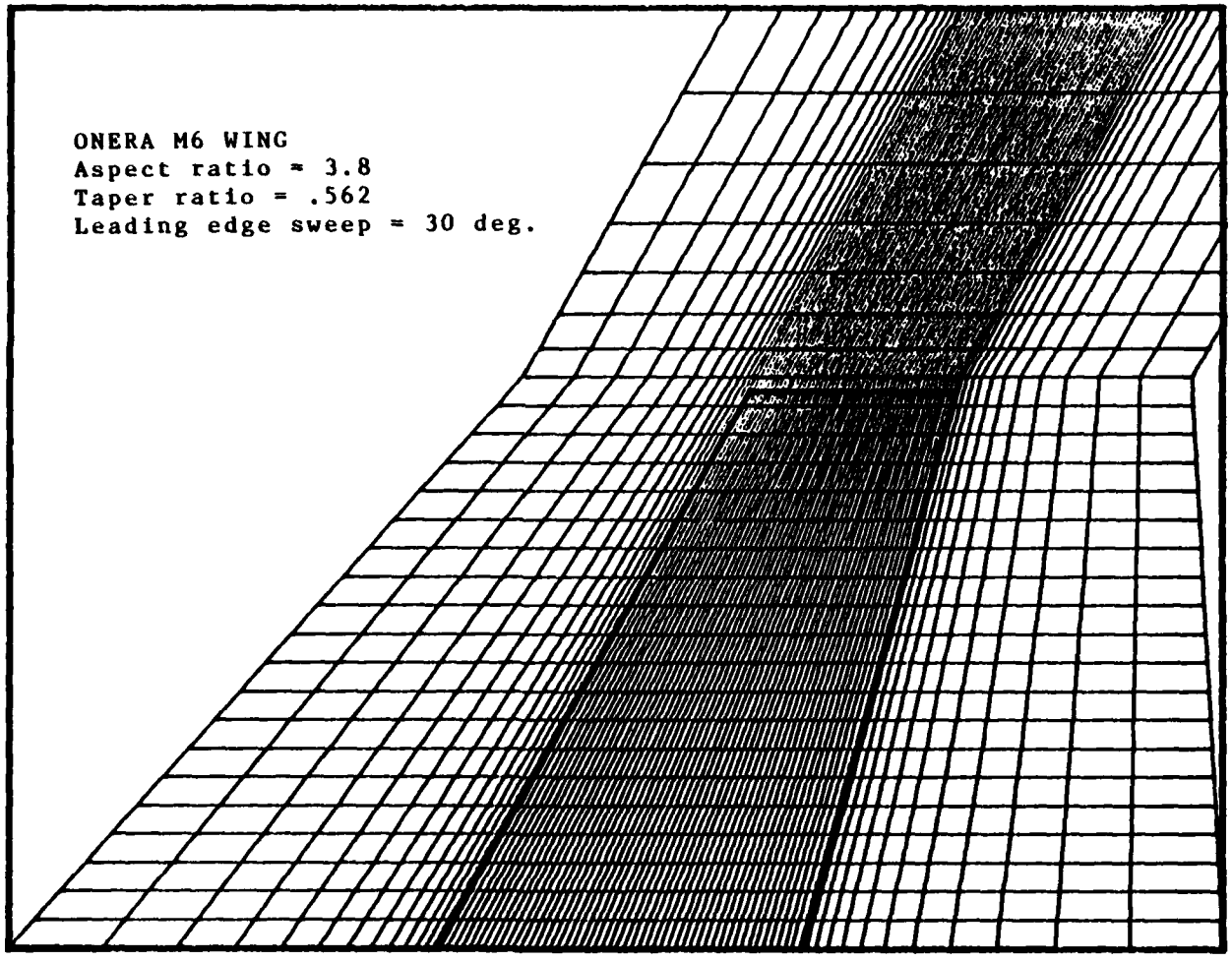
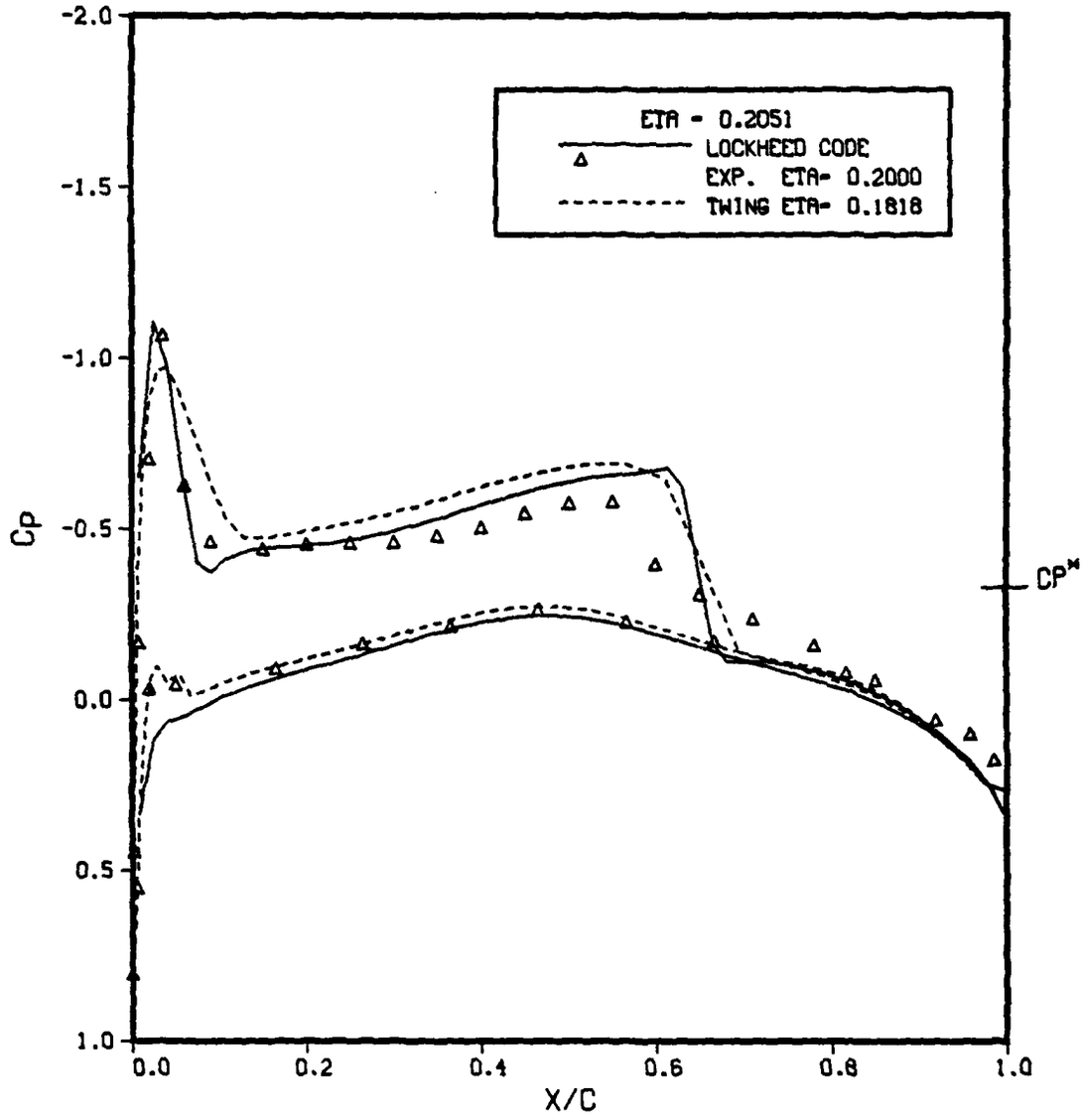


Figure 45. Planform view of sheared cartesian grid



MACH= 0.8395 ALPHA= 3.0600 TWIST= 0.0000

CL = 0.2639 CM = -0.0534 CD = 0.0027

Figure 46. Correlation of pressures from analysis of ONERA M6 wing with experiment and TWING, η = 0.20

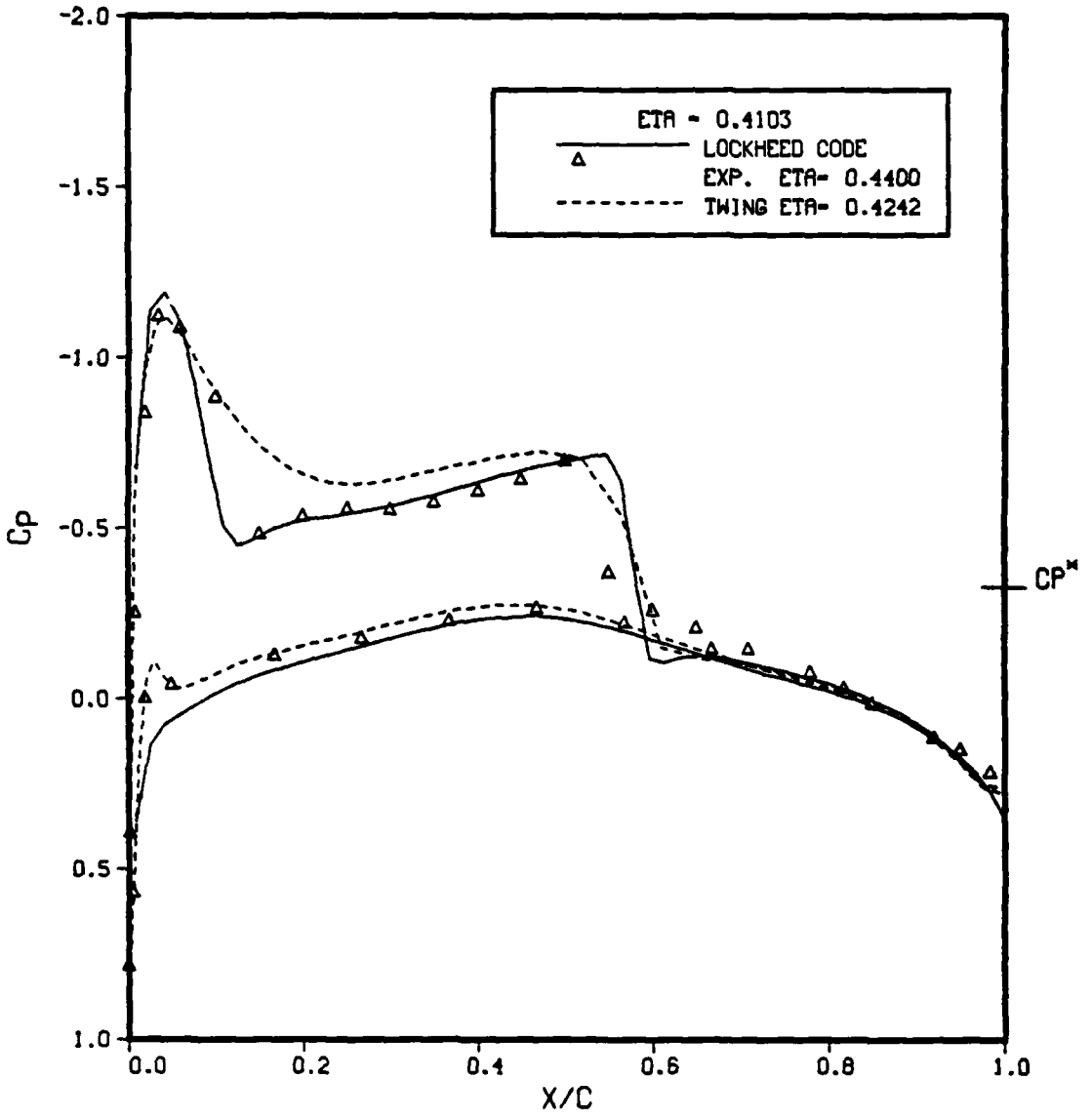


Figure 47. Correlation of pressures from analysis of ONERA M6 wing with experiment and TWING, η = 0.4103

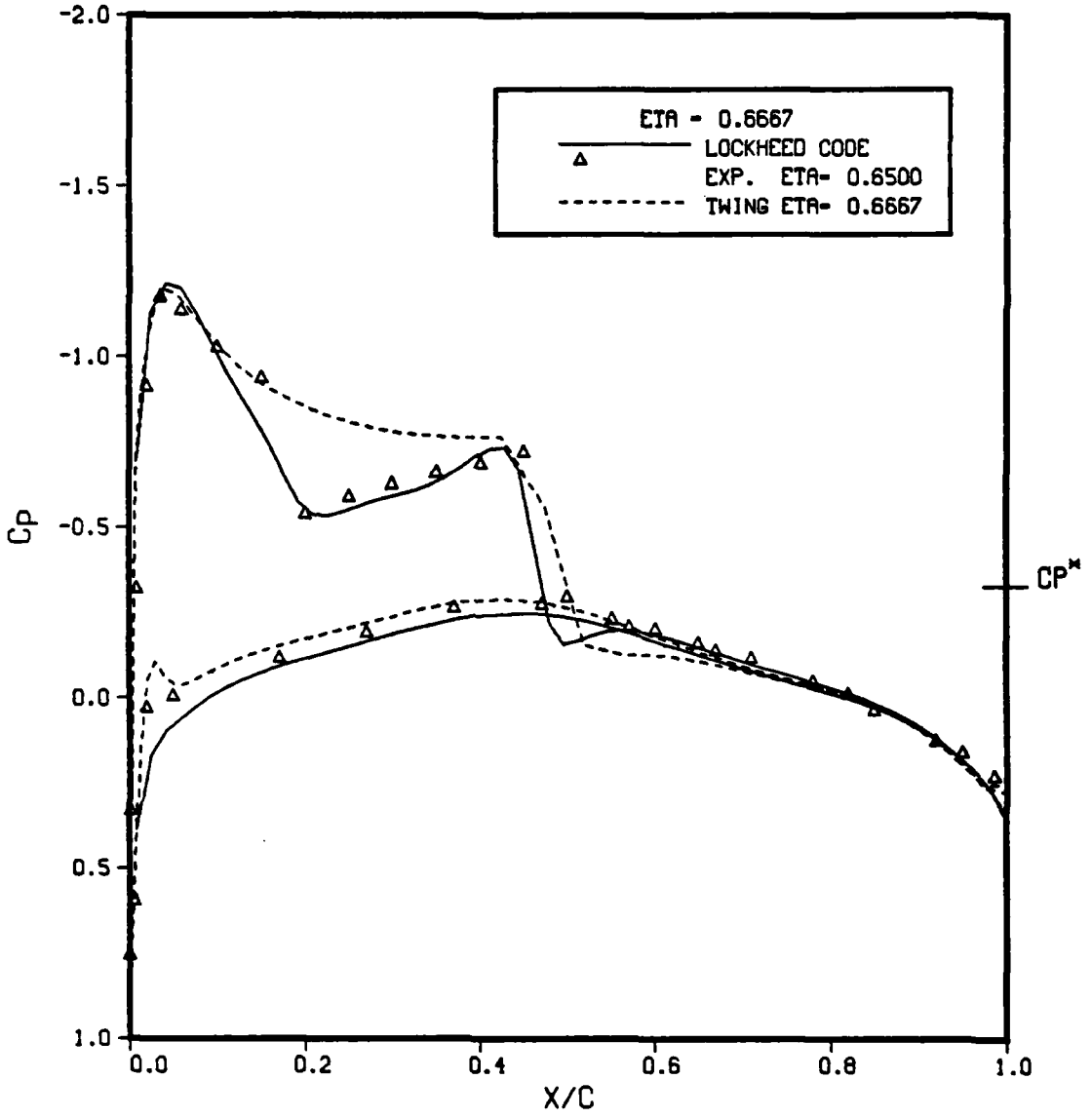


Figure 48. Correlation of pressures from analysis of ONERA M6 wing with experiment and TWING, $\eta = 0.6667$

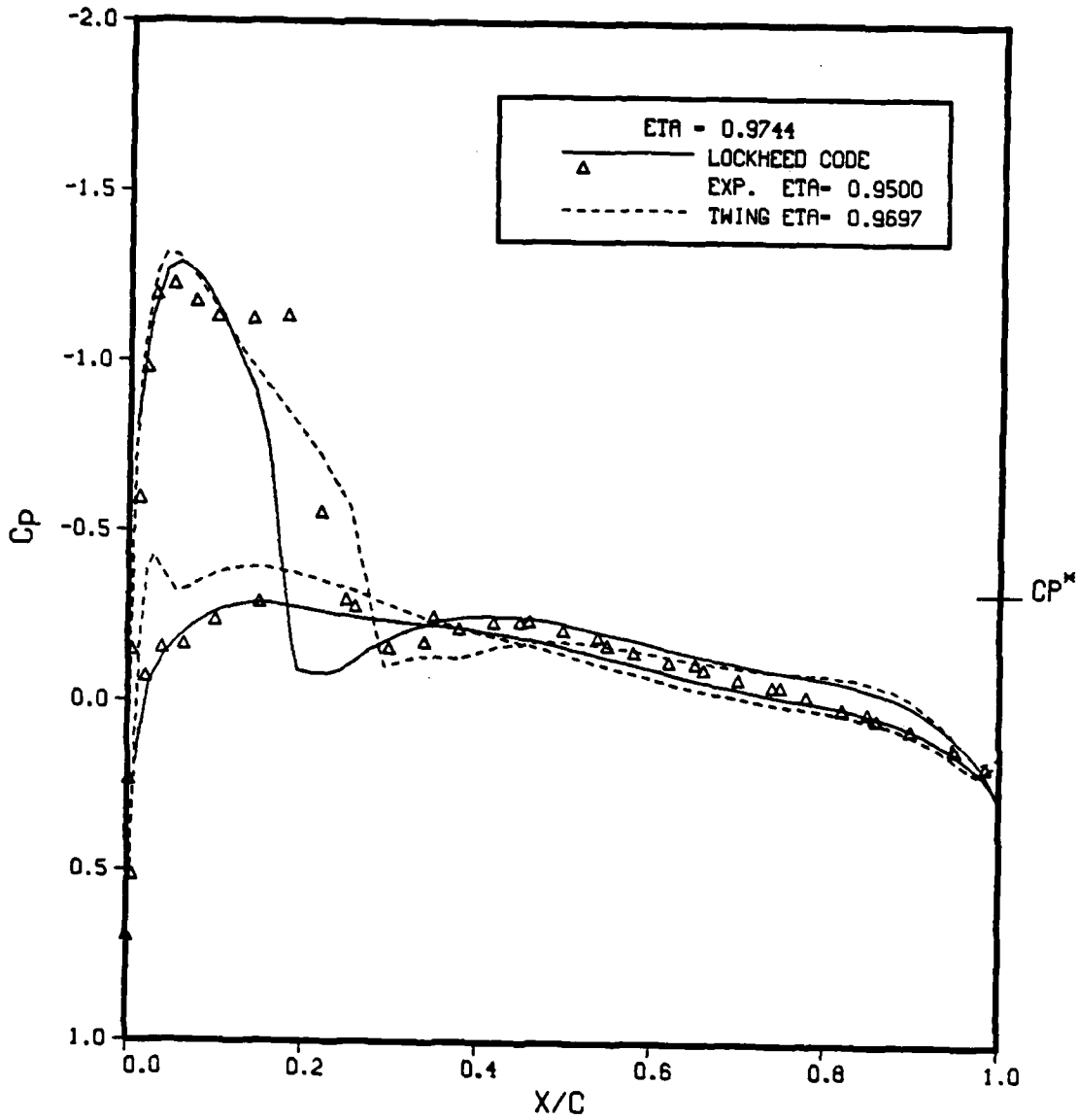


Figure 49. Correlation of pressures from analysis of ONERA M6 wing with experiment and TWING, η = 0.9744

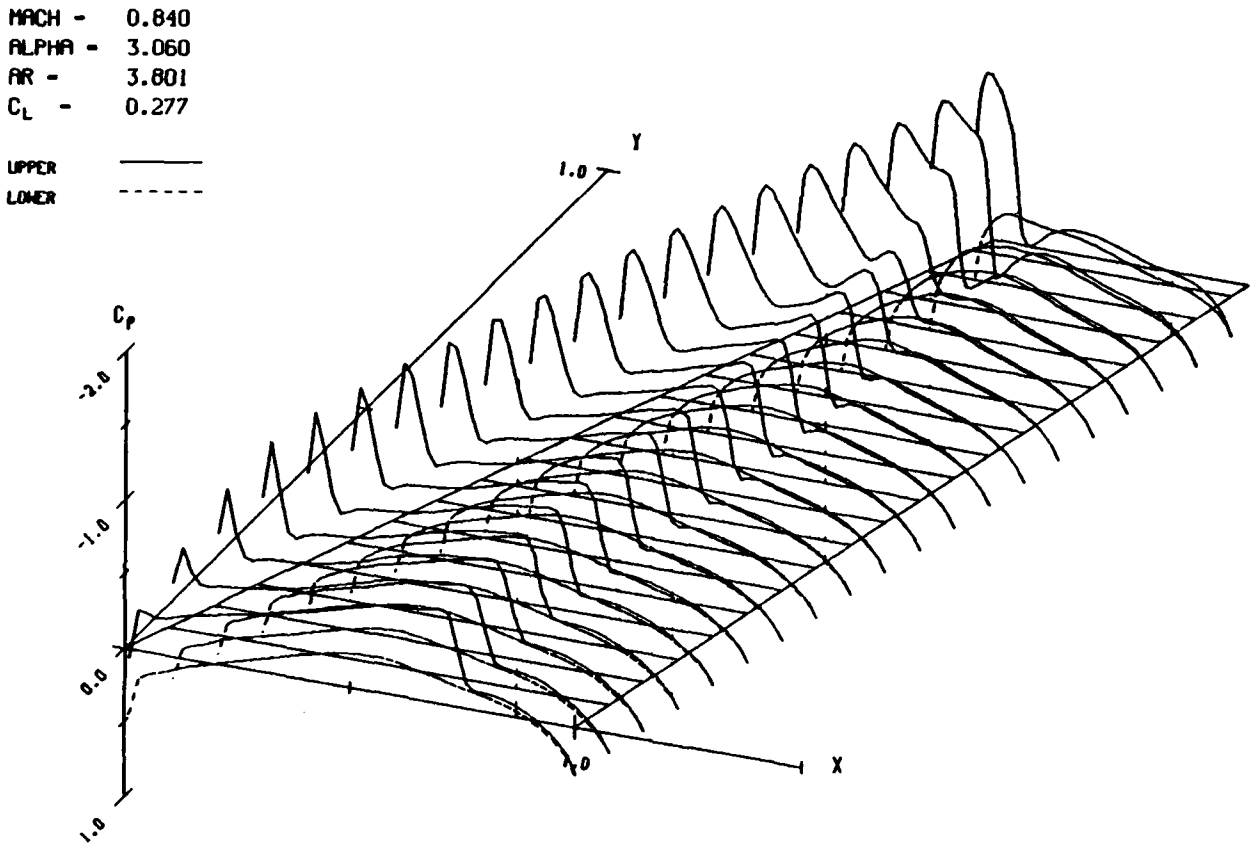


Figure 50. ONERA M6 wing pressure distribution

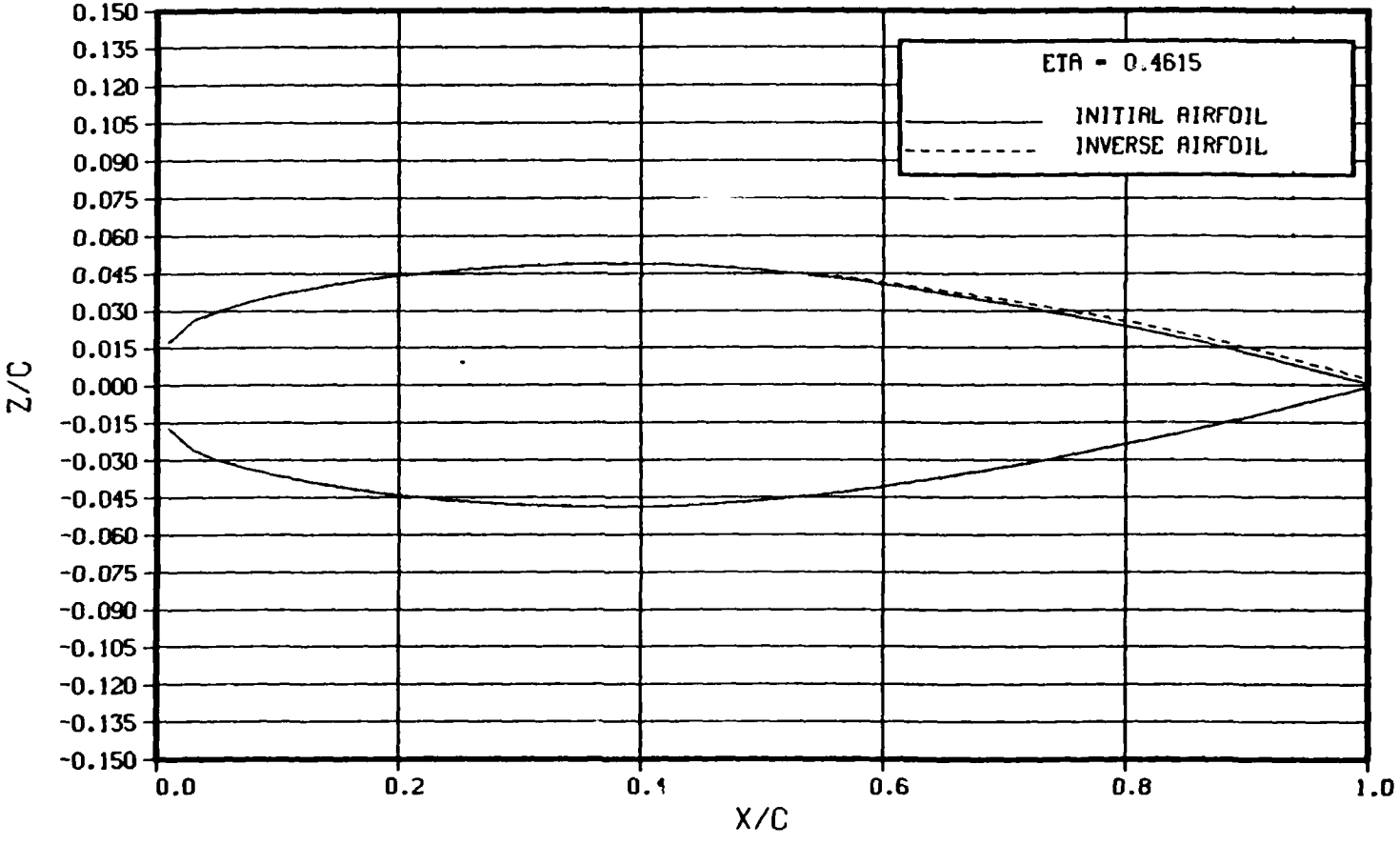


Figure 51. Comparison of ONERA M6 ordinates with ordinates from inverse using analysis target pressures, η = 0.4615

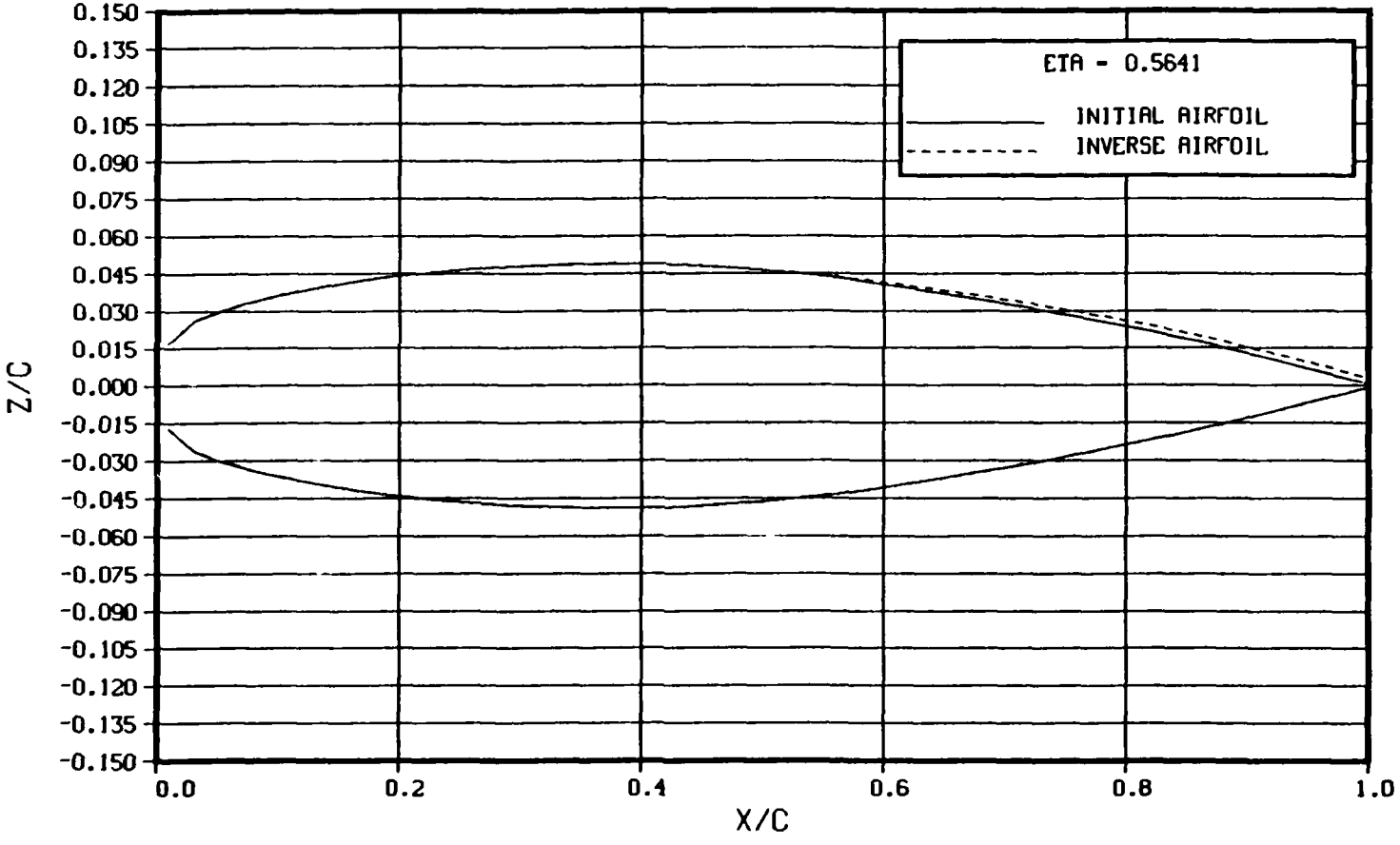


Figure 52. Comparison of ONERA M6 ordinates with ordinates from inverse using analysis target pressures, $\eta = 0.5641$

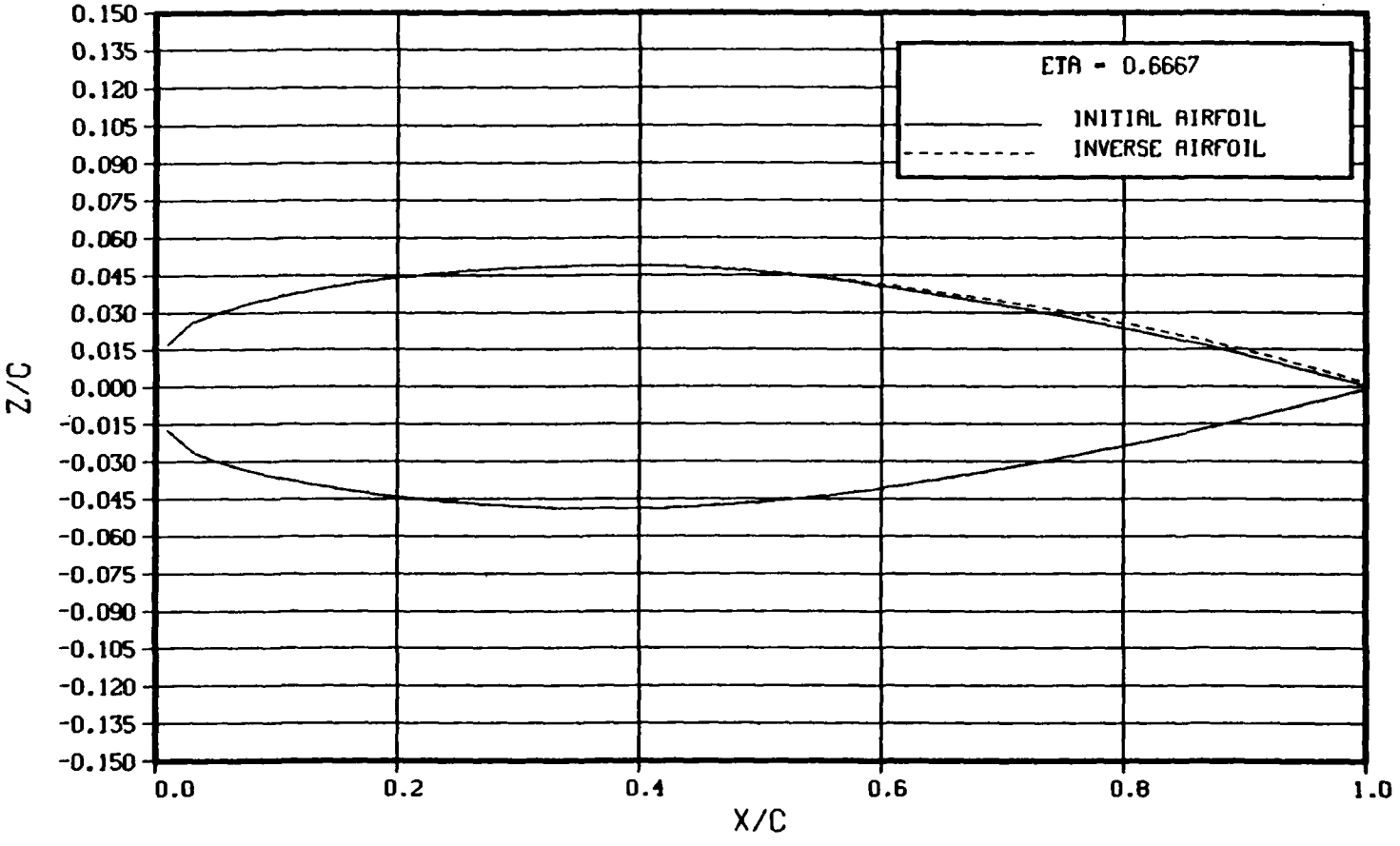


Figure 53. Comparison of ONERA M6 ordinates with ordinates from inverse using analysis target pressures, $\eta = 0.6667$

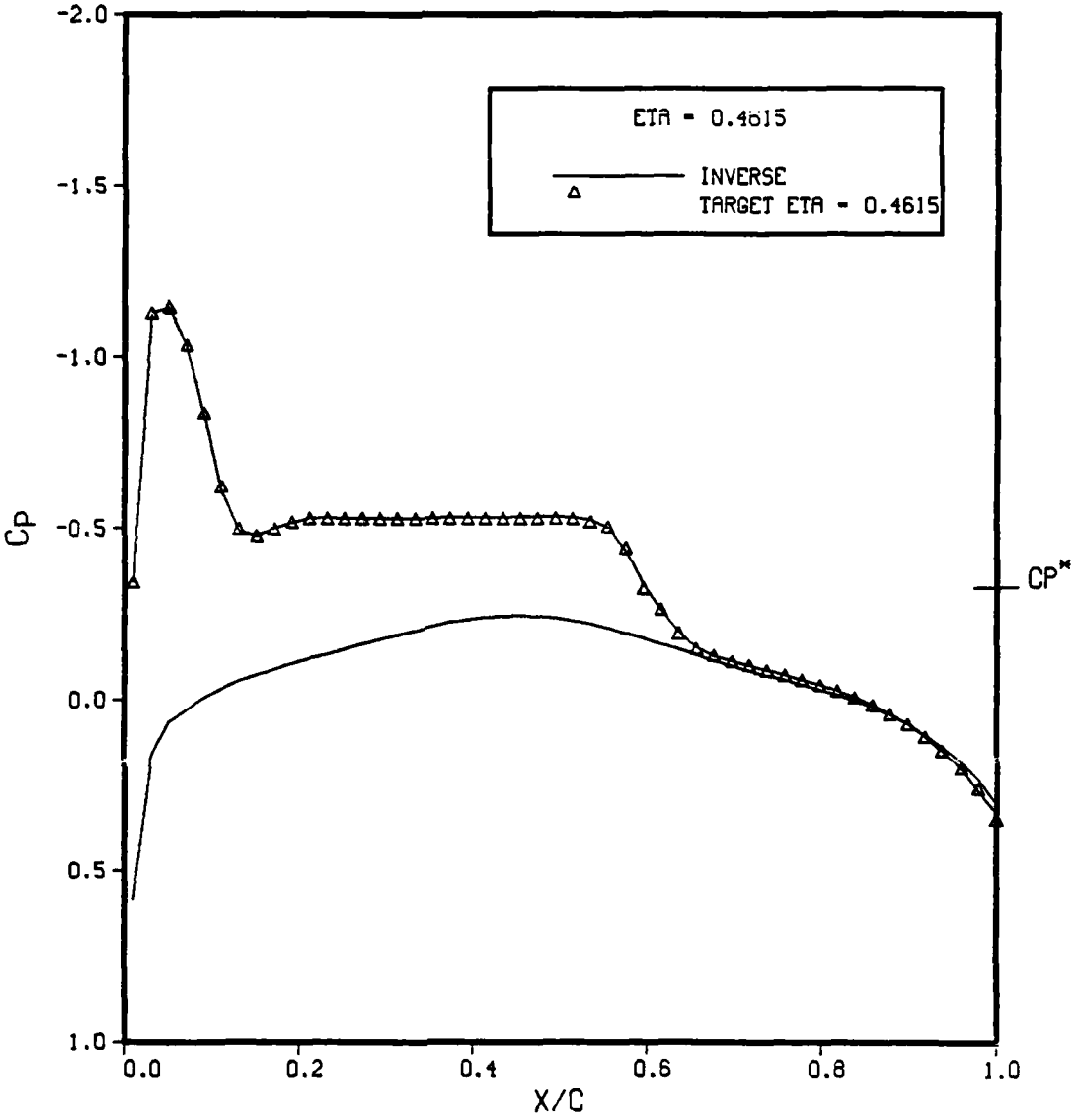


Figure 54. Comparison of inverse pressures with modified target pressures, η = 0.4615

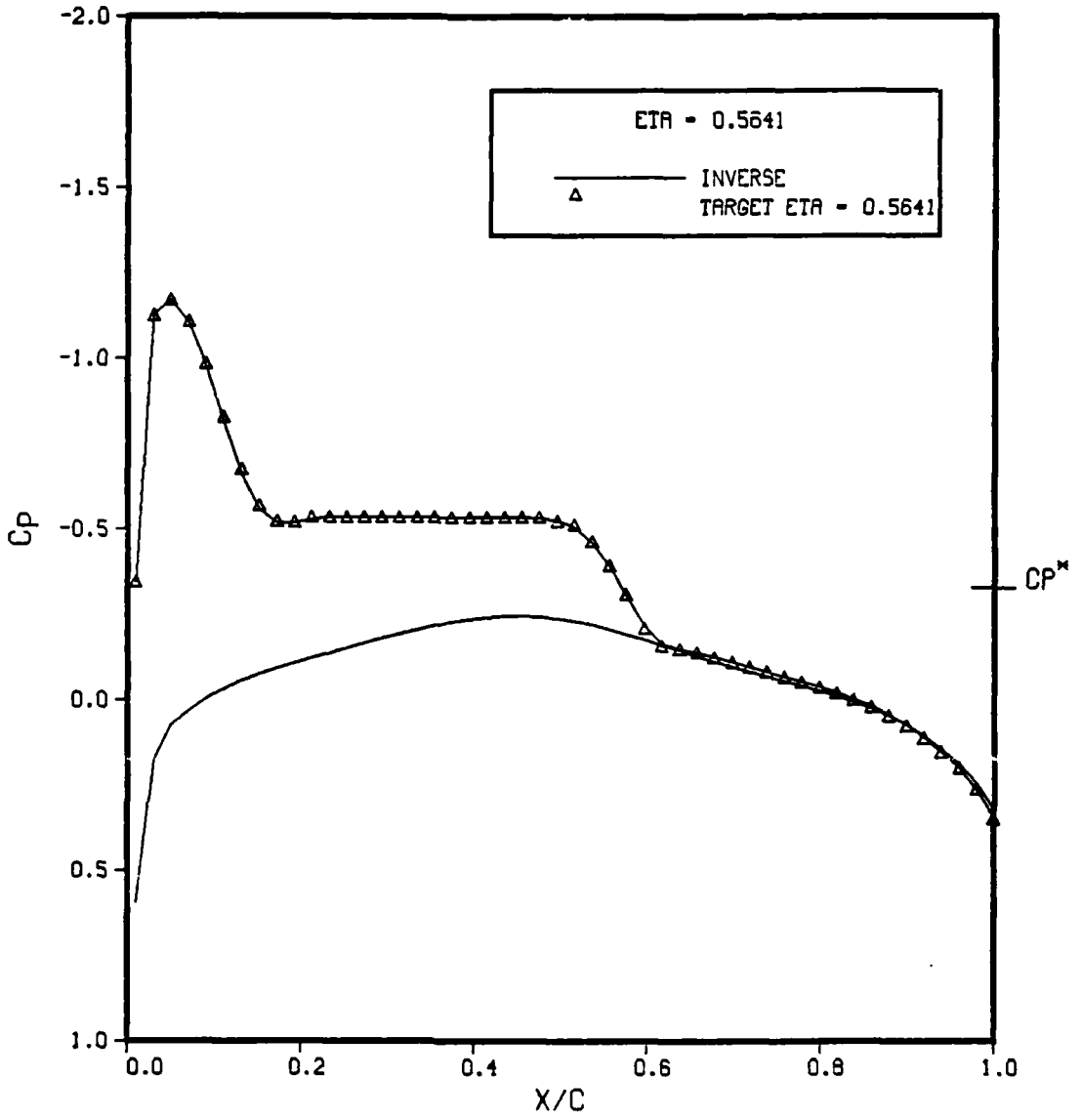


Figure 55. Comparison of inverse pressures with modified target pressures, η = 0.5641

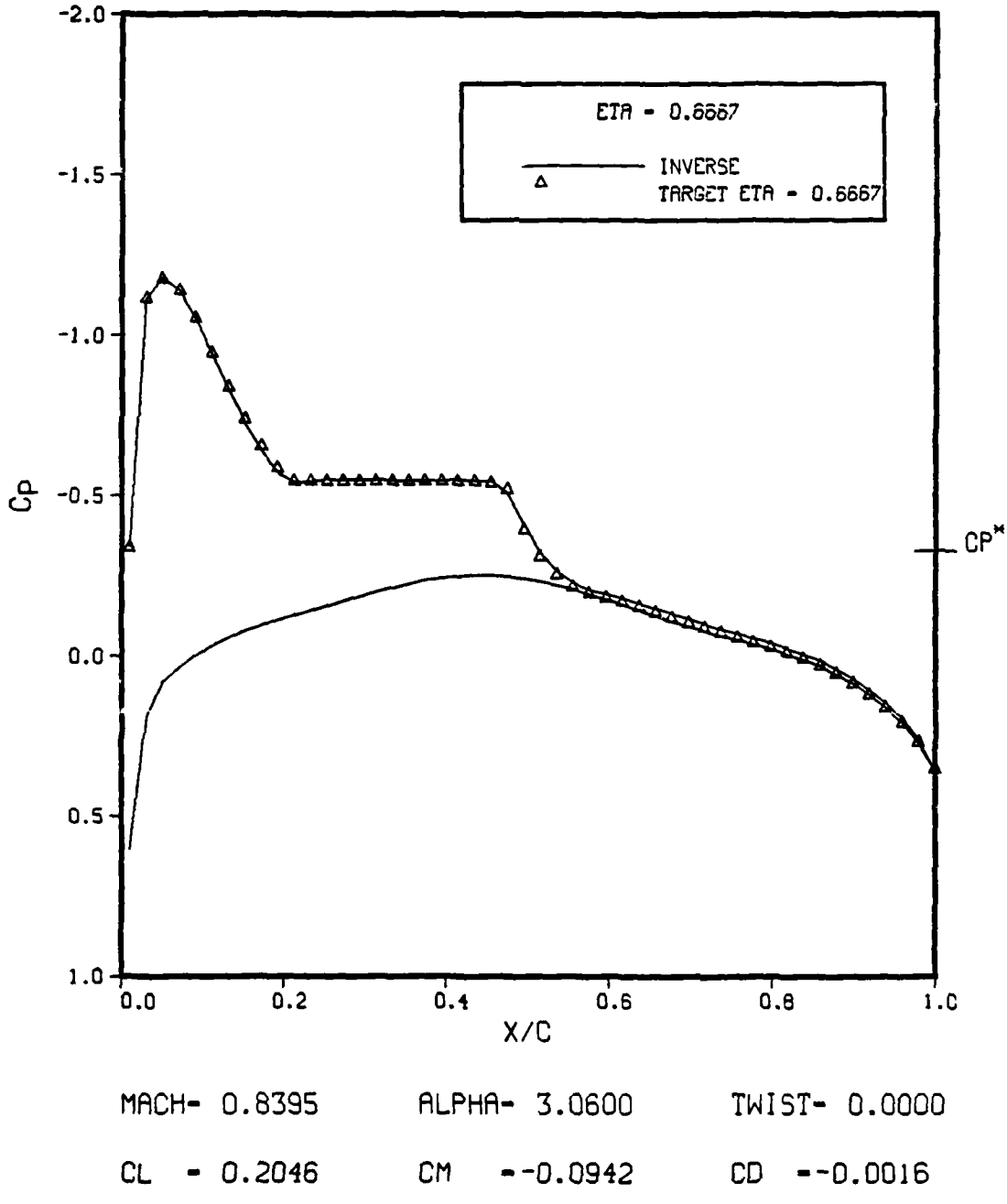


Figure 56. Comparison of inverse pressures with modified target pressures, $\eta = 0.6667$

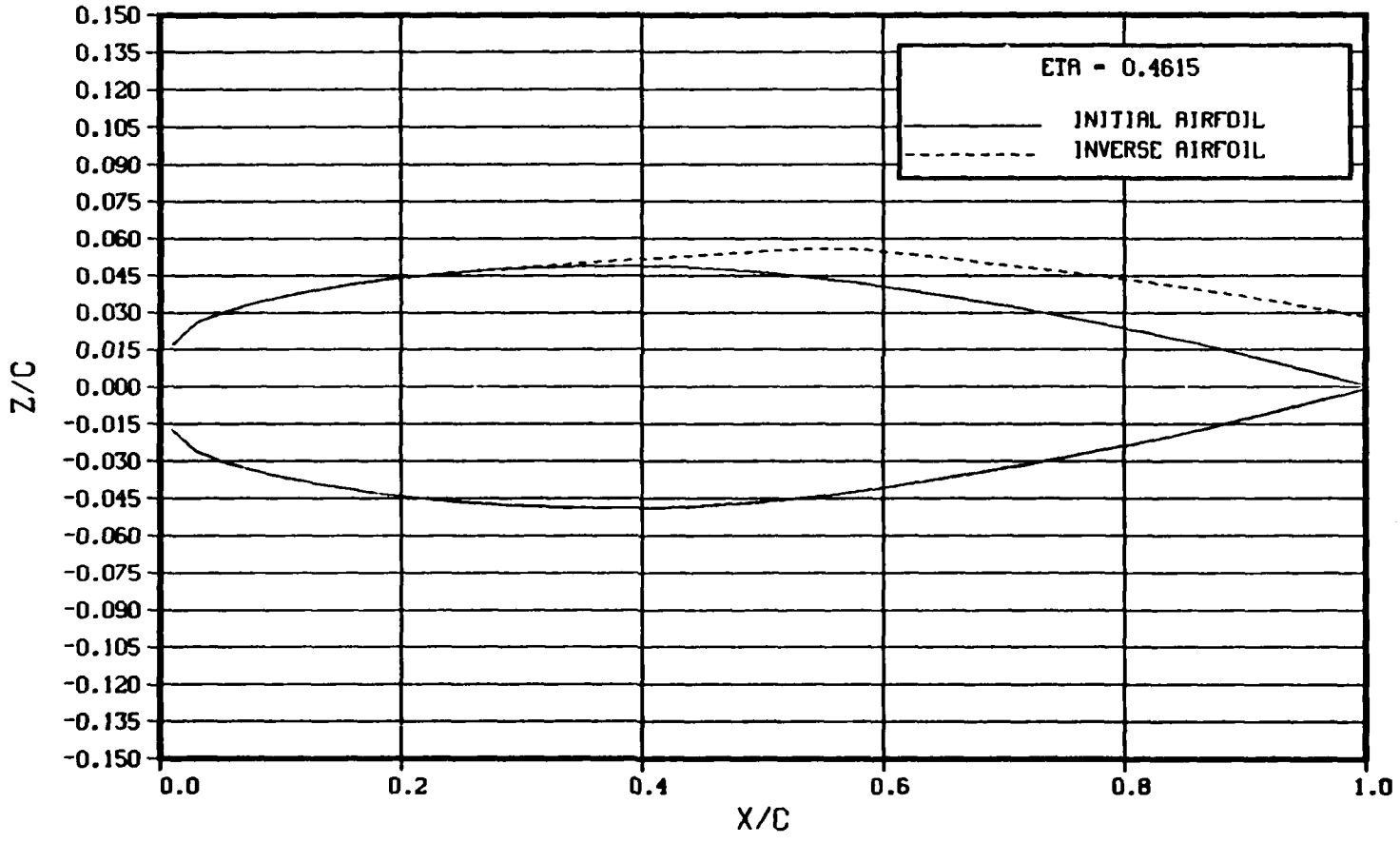


Figure 57. Comparison of ONERA M6 airfoil ordinates and ordinates from inverse using modified target pressures, $\eta = 0.5641$

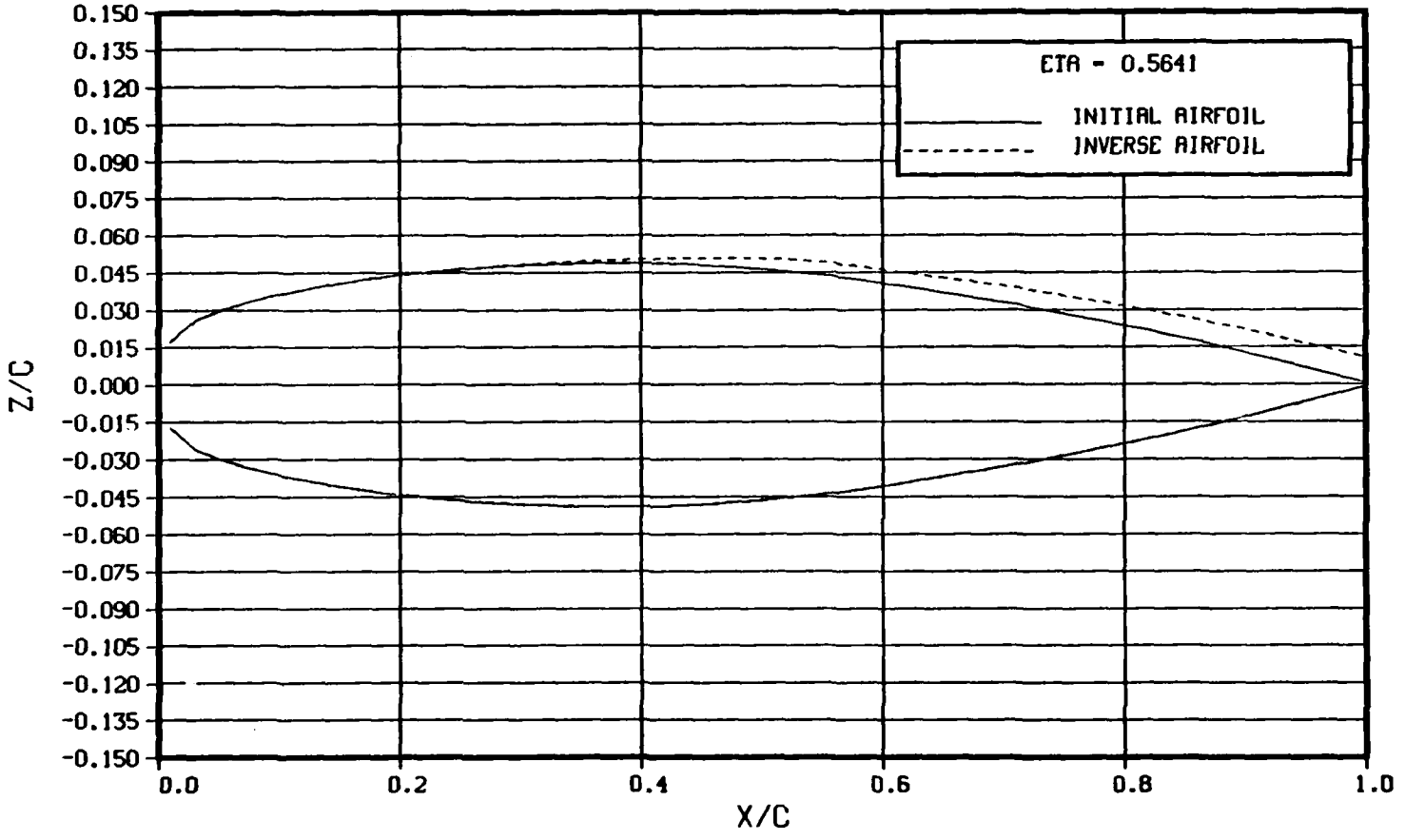


Figure 58. Comparison of ONERA M6 airfoil ordinates and ordinates from inverse using modified target pressures, $\eta = 0.4615$

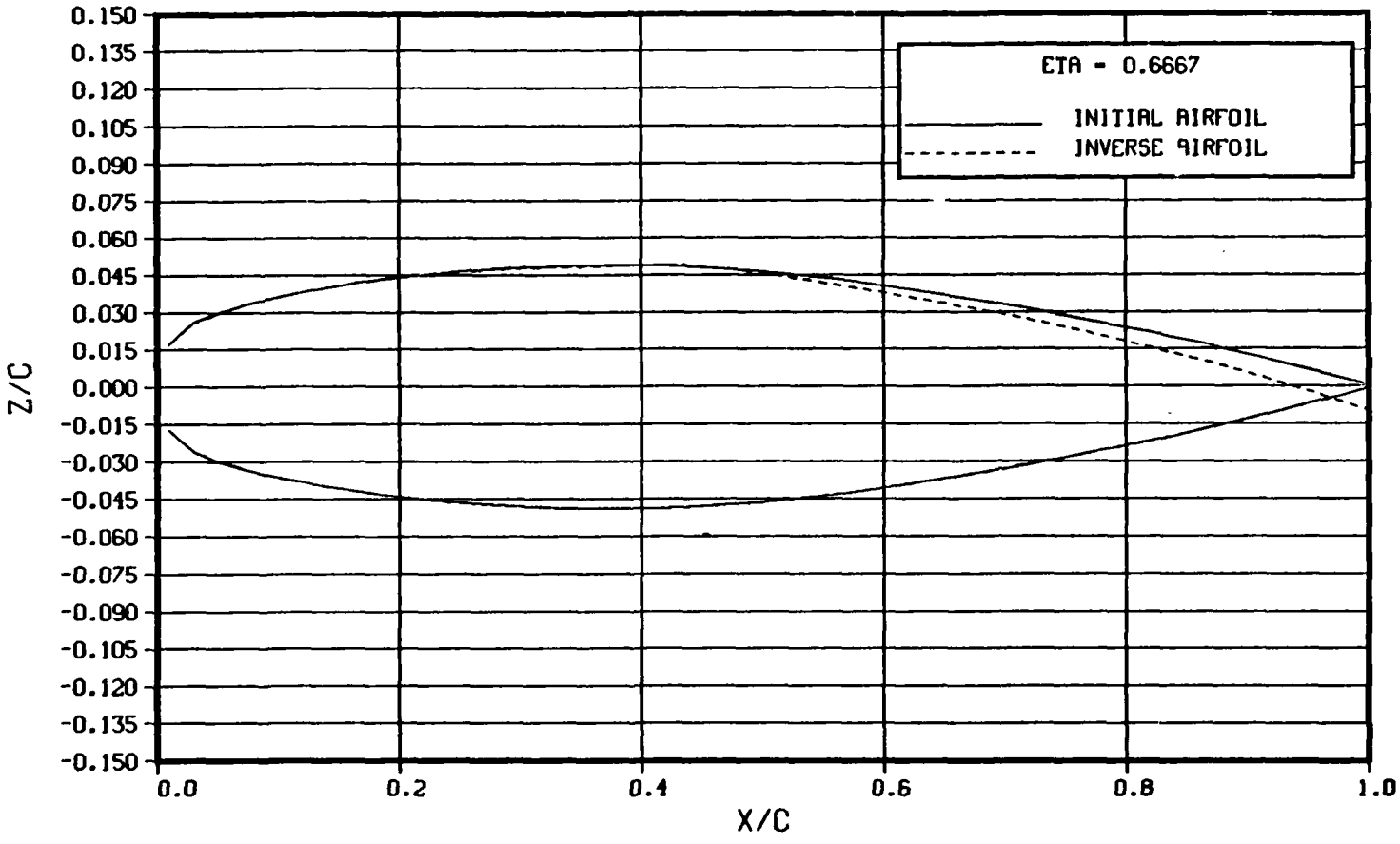
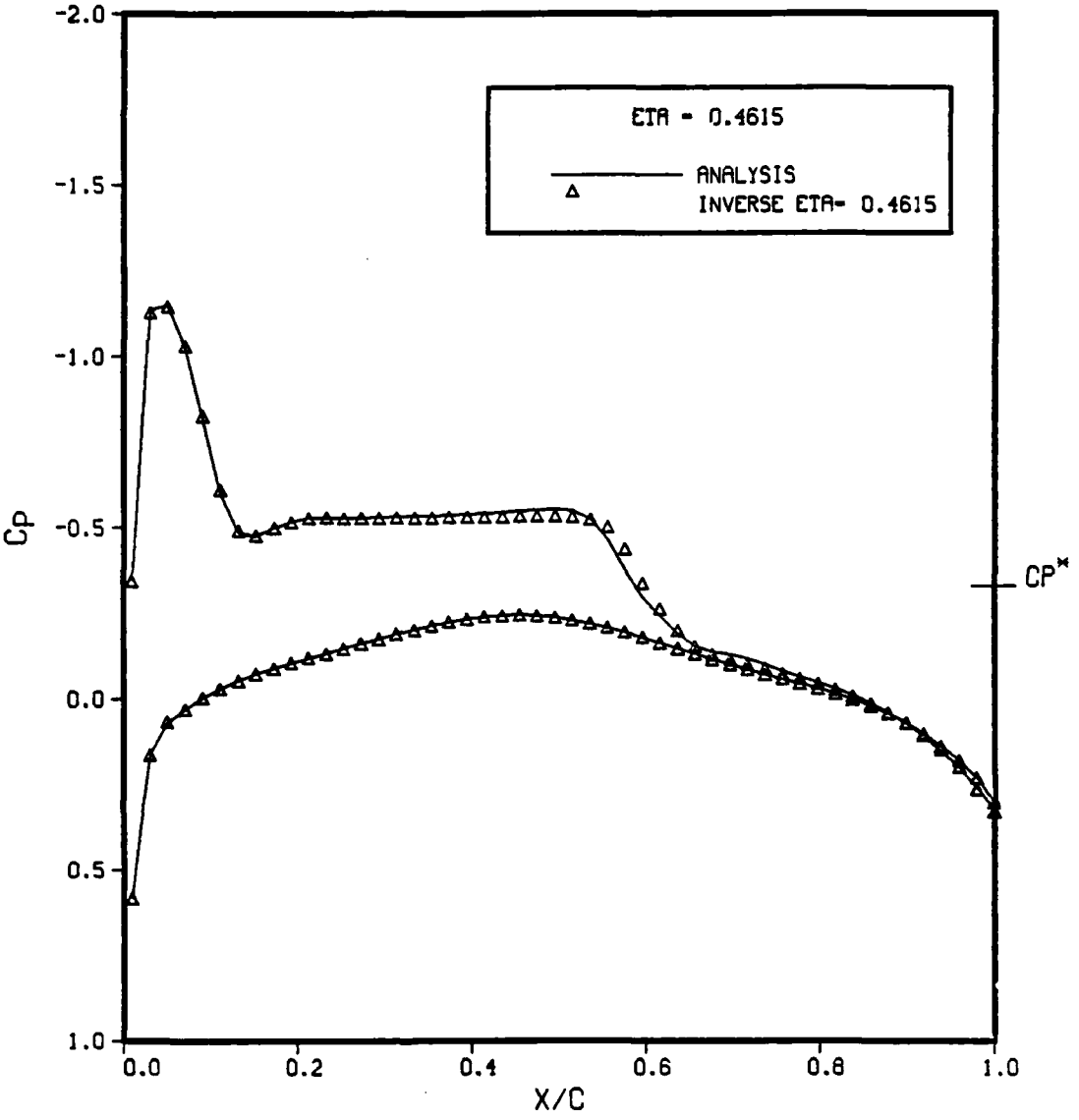


Figure 59. Comparison of ONERA M6 airfoil ordinates and ordinates from inverse using modified target pressures, $\eta = 0.6667$



MACH= 0.8395 ALPHA= 3.0600 TWIST= 0.0000

CL = 0.2233 CM = -0.0732 CD = -0.0043

Figure 60. Correlation of inverse pressures and pressures from analysis of modified wing, η = 0.4615

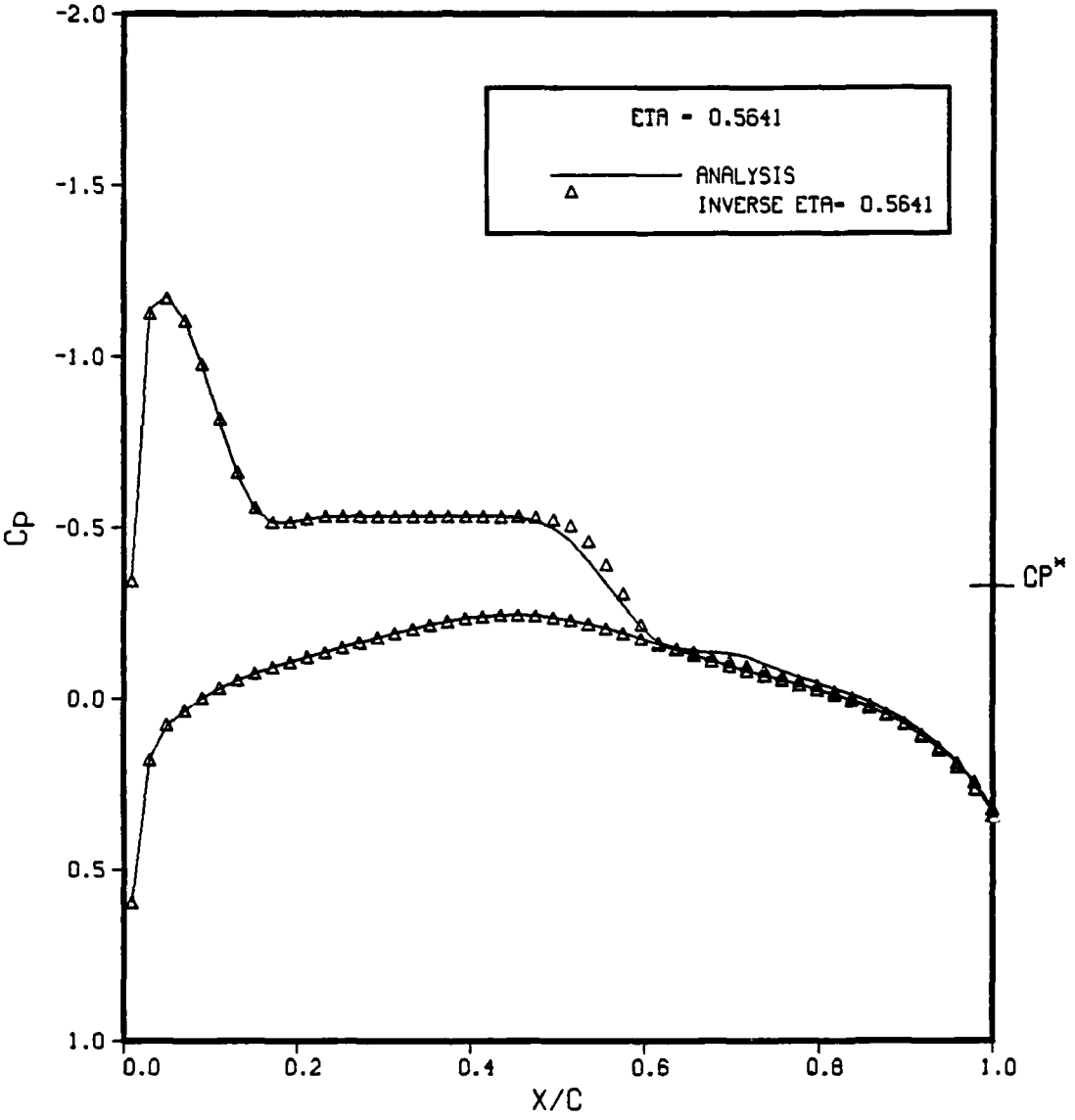


Figure 61. Correlation of inverse pressures and pressures from analysis of modified wing, η = 0.5641

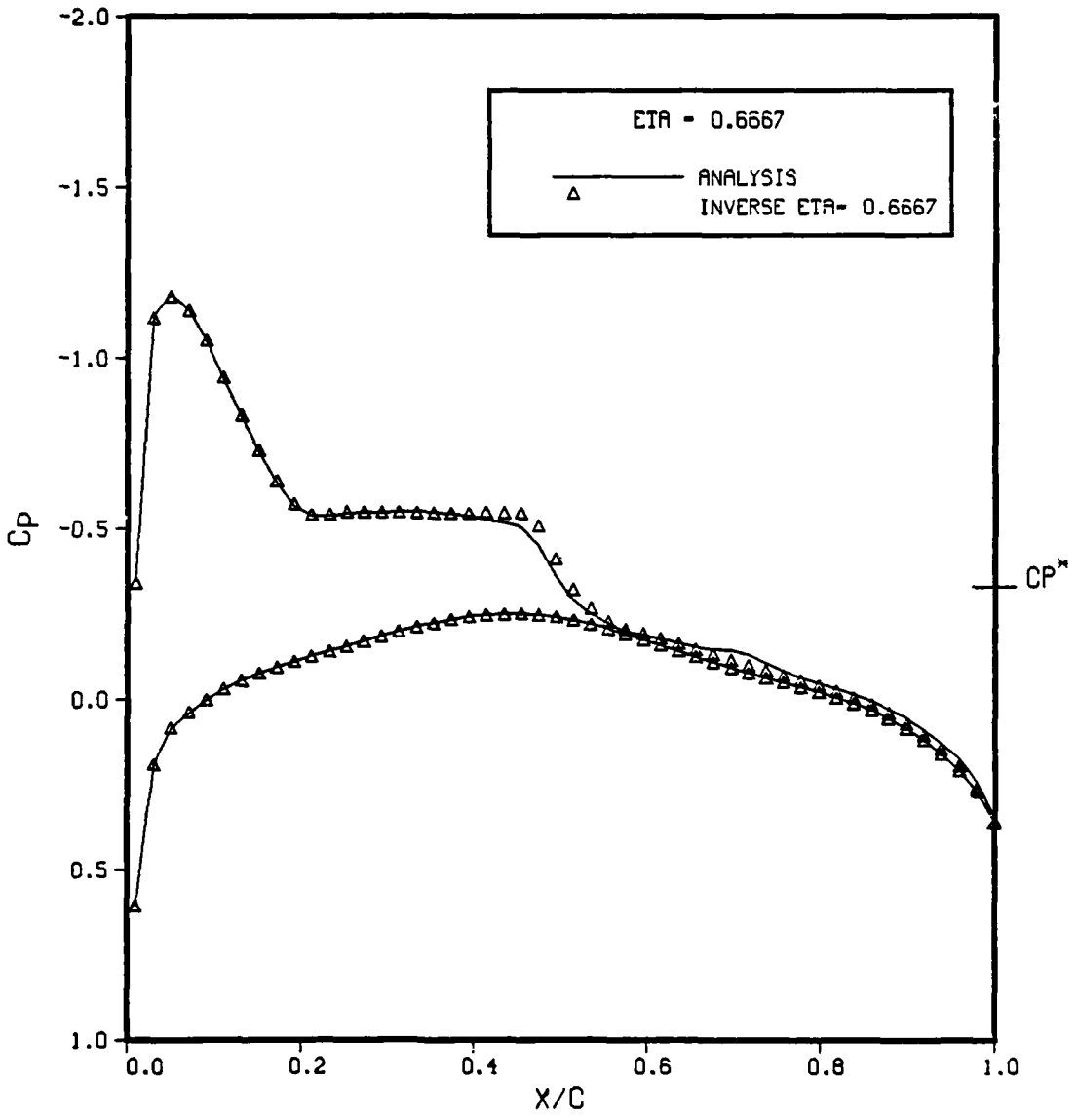


Figure 62. Correlation of inverse pressures and pressures from analysis of modified wing, η = 0.6667

APPENDIX A
ZEBRA II ALGORITHM

The ZEBRA II scheme of South et al.[1-2] solves the conservative form of the full potential equation

$$(\rho u)_x + (\rho v)_y + (\rho w)_z = 0 \tag{A-1}$$

where

$$\begin{aligned} u &= \phi_x \\ v &= \phi_y \\ w &= \phi_z \\ \rho &= (M_\infty^2\, a^2)^{\frac{1}{\gamma-1}} \\ a^2 &= \frac{1}{M_\infty^2} + \frac{\gamma-1}{2}(1 - q^2) \\ q^2 &= u^2 + v^2 + w^2 \end{aligned} \tag{A-2}$$

Equation (A-1) is replaced by its finite difference analog on an evenly spaced grid

$$\bar{\delta}_x\, (\rho\phi_x)_{i+\frac{1}{2},j,k} + \bar{\delta}_y\, (\rho\phi_y)_{i,j+\frac{1}{2},k} + \bar{\delta}_z\, (\rho\phi_z)_{i,j,k+\frac{1}{2}} = 0 \tag{A-3}$$

where $\bar{\delta}_x$, $\bar{\delta}_y$, and $\bar{\delta}_z$ are first-order backwards differences. For example,

$$\bar{\delta}_x\, (\rho\phi_x)_{i+\frac{1}{2},j,k} = \frac{(\rho\phi_x)_{i+\frac{1}{2},jk} - (\rho\phi_x)_{i-\frac{1}{2},jk}}{\Delta x} \tag{A-4}$$

where

$$\begin{aligned} \phi_{x_{i+\frac{1}{2},jk}} &= \frac{\phi_{i+1,jk} - \phi_{i,jk}}{\Delta x} \\ \phi_{x_{i-\frac{1}{2},jk}} &= \frac{\phi_{i,jk} - \phi_{i-1,jk}}{\Delta x} \end{aligned} \tag{A-5}$$

The density ρ is replaced by the upwinded artificial compressibility value $\bar{\rho}$ to stabilize calculations in regions of supersonic flow. The value

of $\bar{\rho}$ is given by

$$\bar{\rho} = \rho - \mu \left[\frac{u}{q_\infty} \Delta x \frac{\partial \rho}{\partial x} + \frac{v}{q_\infty} \Delta y \frac{\partial \rho}{\partial y}\right] \tag{A-6}$$

where $\partial\rho/\partial x$ and $\partial\rho/\rho y$ are upwind differenced at supersonic points, Δx is the chordwise grid spacing and Δy is the spanwise grid spacing. The upwind switching function μ is given by

$$\mu = \text{MAX}\ (0, 1 - \frac{a^2}{q^2}) \tag{A-7}$$

In the present ZEBRA II code v/q_∞ and $\partial\rho/\partial y$ are assumed to be negligible and u/q_∞ is assumed to be approximately one. Equation (A-6) then becomes

$$\bar{\rho} = \rho - \mu \frac{\partial \rho}{\partial x} \tag{A-8}$$

The ZEBRA II algorithm solves Eq. (A-3) using an iterative scheme that mimics point Successive Overrelation (SOR). For 3-D calculations, the ZEBRA II algorithm marches in the streamwise (I) direction solving one spanwise plane at a time. In each plane, points J+K odd are denoted black and points J+K even are termed white. Each plane is solved by a two-pass sweep in which new black values are obtained first, followed by the white points. In this way, convergence is accelerated because calculations at the white points will use updated quantities at the black points.

By replacing ϕ in Eq. (A-3) with

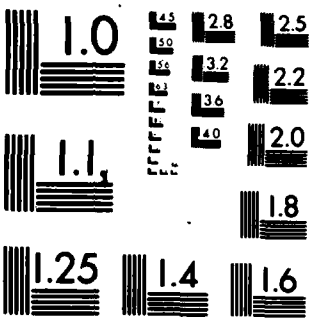
$$\bar{\phi}^{N+1} = \frac{\Delta\phi}{\omega} + \phi^N \tag{A-9}$$

where $\Delta\phi = \phi^{N+1} - \phi^N$ is the correction, ω is an acceleration parameter, and N is the iteration number. The solution at each grid point is given by

$$\Delta\phi_{i,j,k} = \frac{1}{B}\left[\frac{\Delta x^2 R_{i,j,k}}{\bar{\rho}_{ave}} + \beta\Delta\phi_{i-1,j,k}\right] \tag{A-10}$$

AD-A129 573 COMBINED DIRECT/INVERSE THREE-DIMENSIONAL TRANSONIC WING DESIGN(U) LOCKHEED-GEORGIA CO MARIETTA R A WEED ET AL. MAY 83 LG83-ER-0060 2/2

UNCLASSIFIED DTNSRDC/ASED-CR-03-83 N00167-81-C-0078 F/G 20/4 NL



where

$$B = \frac{2}{\omega_x} + \frac{2}{\omega_y}\left(\frac{\Delta x}{\Delta y}\right)^2 + \frac{2}{\omega_z}\left(\frac{\Delta x}{\Delta z}\right)^2 + \beta \qquad \text{(A-11)}$$

The $\beta\Delta\phi_{i-1}$ term in Eq. (A-10) comes from the inclusion of a ϕ_{xt} term to add explicit temporal damping to the algorithm; β is the damping coefficient. It should be noted that the ZEBRA II algorithm possesses some natural temporal damping since the points in the I+1 plane are not as updated as the points in the I-1 plane. In addition, the black points in each I plane are not as current as the white points.

The residual, $\Delta x^2 R_{i,j,k}$ in Eq. (A-10) can be written

$$\Delta x^2 R_{i,j,k} = \bar{\rho}_{i+\frac{1}{2},j,k}\left(\phi^N_{i,j+1,k} - \phi^N_{i,j,k}\right) - \bar{\rho}_{i-\frac{1}{2},j,k}\left(\phi^N_{i,j,k} - \phi^{N+1}_{i-1,j,k}\right) +$$

$$\left[\bar{\rho}_{i,j+\frac{1}{2},k}\left(\phi^\nu_{i,j+1,k} - \phi^N_{i,j,k}\right) - \bar{\rho}_{i,j-\frac{1}{2},k}\left(\phi^N_{i,j,k} - \phi^\nu_{i,j-1,k}\right)\right]\left(\frac{\Delta x}{\Delta y}\right)^2 +$$

$$\left[\bar{\rho}_{i,j,k+\frac{1}{2}}\left(\phi^\nu_{i,j,k+1} - \phi^N_{i,j,k}\right) - \bar{\rho}_{i,j,k-\frac{1}{2}}\left(\phi^N_{i,j,k} - \phi^\nu_{i,j,k-1}\right)\right]\left(\frac{\Delta x}{\Delta z}\right)^2 \qquad \text{(A-12)}$$

$\nu = N$ for J+K odd

$\nu = N+1$ for J+K even

APPENDIX B

DERIVATION OF PRESSURE COEFFICIENT BOUNDARY CONDITION FOR BASELINE UNIFIED DESIGN/ANALYSIS CODE

Normalizing the full potential equation by the critical speed of sound and the stagnation density leads to a slightly different equation for the pressure coefficient than that used in the TAMU pilot code. The coefficient of pressure can be written

$$C_p = \frac{2}{\gamma M_\infty^2} \left[\frac{P}{P_s} \frac{P_s}{P_\infty} - 1\right] \tag{B-1}$$

where P is the local pressure, P_S is the stagnation pressure, and P_∞ is the free stream pressure. Thus,

$$\frac{P_s}{P_\infty} = \left[1 + \frac{\gamma-1}{2} M_\infty^2\right]^{\frac{\gamma}{\gamma-1}} \tag{B-2}$$

$$\frac{P}{P_s} = \left(\frac{\rho}{\rho_s}\right)^{\gamma} \tag{B-3}$$

$$\frac{\rho}{\rho_s} = \left[1 - \frac{\gamma-1}{\gamma+1} (\phi_x^2 + \phi_y^2 + \phi_z^2)\right]^{\frac{1}{\gamma-1}} \tag{B-4}$$

Now let $P_S/P_\infty = C_1$ and

$$\phi_x^2 + \phi_y^2 + \phi_z^2 = \phi_x^2 \left(1 + \frac{\phi_y^2}{\phi_x^2} + \frac{\phi_z^2}{\phi_x^2}\right) \tag{B-5}$$

Substituting into Eq. (B-1) yields

$$\frac{\gamma M_\infty^2}{2} C_p + 1 = C_1 \left[1 - \frac{\gamma-1}{\gamma+1} \phi_x^2 \left(1 + \frac{\phi_y^2}{\phi_x^2} + \frac{\phi_z^2}{\phi_x^2}\right)\right]^{\frac{\gamma}{\gamma-1}} \tag{B-6}$$

Solving Eq. (B-6) yields

$$\phi_x^2 = \frac{\frac{\gamma+1}{\gamma-1}\left[1 - \left(\frac{\gamma M_\infty^2 C_p}{2C_1} + \frac{1}{C_1}\right)^{\frac{\gamma-1}{\gamma}}\right]}{\left[1 + \left(\frac{\phi_y^2}{\phi_x^2}\right) + \left(\frac{\phi_z^2}{\phi_x^2}\right)\right]} \qquad \text{(B-7)}$$

The procedure outlined in the description of the development of the three-dimensional inverse pilot code is used to extract a value of ϕ as a function of C_p from Eq. (B-7) for use as a Dirichlet boundary condition in the inverse design scheme.

REFERENCES

1. South, J. C., Keller, J. D., and Hafez, M., "Vector Processor Algorithms for Transonic Flow Calculations," AIAA Journal, Vol. 18, No. 7, 1980, pp. 786-792.

2. South, J. C., Keller, J. D., and Hafez, M., "Computational Transonics on a Vector Computer," ARO Report 80-3, Proceedings of 1980 Army Numerical Analysis and Computers Conference, March 1980, pp. 357-368.

3. Jameson, A., "Iterative Solution of Transonic Flows over Airfoils and Wings, Including Flows at Mach 1," Communications On Pure And Applied Math., Vol. 27, 1974, pp. 283-309.

4. Holst, T. L., "Fast Conservative Algorithm for Solving the Transonic Full Potential Equation," AIAA Paper 79-1456, July 1979; also AIAA Journal, Vol. 18., No. 12, Dec. 1980, pp. 1431-1439.

5. Holst, T. L., "An Implicit Algorithm for Solving the Transonic, Conservative Full Potential Equation", ARO Report 80-3, Proceedings of the 1980 Army Numerical Analysis and Computers Conference, March 1980, pp. 197-222.

6. Holst, T. L., and Thomas, S., "Numerical Solution of Transonic Wing Flow Fields," AIAA Paper 82-0105, January 1982.

7. Purcell, T. W. and Carlson, L. A., "Transonic Airfoil Design and Analysis Using the Full Potential Equation and Approximate Boundary Conditions," Texas A&M Research Foundation Report TAMRF-4276-8101, August 1981.

8. Anderson, W. K. and Carlson, L. A., "Inverse Wing Design on a Vector Computer," Texas A&M Research Foundation Report TAMRF-4535-8218, December 1982.

9. Boppe, C. W., "Transonic Flow field Analysis for Wing-Fuselage Configurations," NASA CR-3243, 1980.

10. Bailey, F. R. and Ballhaus, W. F., "Comparison of Computed and Experimental Pressures for Transonic Flows about Isolated Wings and Wing-Fuselage Configurations, NASA SP-347, 1975, pp 1213-1231.

11. Klunker, E. B., "Contributions to Methods for Calculating the Flow about Thin Lifting Wings at Transonic Speeds - Analytical Expressions for the Far Field," NASA TN D-6530, 1971.

12. Schmitt, V., and Charpin, D., "Pressure Distributions on the ONERA M6 Wing at Transonic Mach Numbers," Experimental Data Base for Computer Program Assessment, Appendix B1, AGARD-AR-138, May 1979.

13. Carlson, L. A., "Transonic Airfoil Design Using Cartesian Coordinates," NASA CR-2578, 1976.

14. Carlson, L. A., "Transonic Airfoil Analysis and Design Using Cartesian Coordinates," Proceedings of AIAA 2nd Computational Fluid Dynamics Conference, June 1975, pp. 175-183.

15. Shankar, V., "A Full Potential Inverse Method For Wing Design Based on a Density Linearization Scheme," NASA CR-165991, October 1982.

16. Shankar, V., "Computational Transonic Inverse Procedure for Three-Dimensional Wing Design with Automated Trailing Edge Closure," AIAA Paper No. 80-1390, July 1980.

17. Vanderplaats, G. N., Hicks, R. N., and Murman, E. M., "Application Of Numerical Optimization Techniques to Airfoil Design," NASA SP-347, 1975.

END

DATE
FILMED

7—83

DTIC