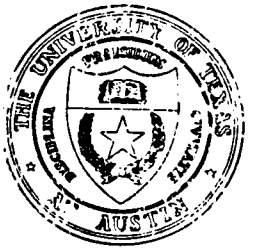
Potential Flow Around Two-Dimensional Airfoils Using A Singular Integral Method

by

Yves Nguyen and Dennis Wilson
Mechanical Engineering Department

Report No. 87-104



THE FLUID DYNAMICS GROUP
BUREAU OF ENGINEERING RESEARCH
THE UNIVERSITY OF TEXAS AT AUSTIN
AUSTIN, TX 78712
December 1987

Potential Flow Around Two-Dimensional Airfoils Using A Singular Integral Method

by

Yves Nguyen and Dennis Wilson
Mechanical Engineering Department

Report No. 87-104

ABSTRACT

The problem of potential flow around two-dimensional airfoils is solved by using a new singular integral method. The potential flow equations for incompressible potential flow are written in a singular integral equation. This equation is solved at N collocation points on the airfoil surface. A unique feature of this method is that the airfoil geometry is specified as an independant variable in the exact integral equation.

Compared to other numerical methods, the present calculation procedure is much simpler and gives remarkable accuracy for many body shapes. An advantage of the present method is that it allows the inverse design calculation and the results are extremely accurate. Compared to other previous calculations, the present design solution is simpler, more accurate and does not use an iteration procedure.

TABLE OF CONTENTS

CHAPTER 1

REVIEW OF EXISTING METHODS

1.1 Introduction

A potential flow is one which is inviscid and irrotational. The irrotational condition implies that the velocity can be defined in terms of a potential function by:

$$\vec{U}=\nabla\phi \tag{1.1}$$

When the problem involves a prescribed free stream flow over an arbitrary body, the velocity is commonly expressed as:

$$\vec{U}=\vec{U}_{\infty}+\vec{q}\,, \tag{1.2}$$

where U_{∞} is the onset flow present when the body is not present and q is the disturbance velocity. In most cases U_{∞} is a uniform flow defined as parallel to the x-axis. When the flow is potential and incompressible, the Navier-Stokes equations reduce to the following equation:

$$\nabla^2\phi = 0 \tag{1.3}$$

The flow field is completely determined by kinematics, when the appropriate boundary conditions are specified. On the surface of the airfoil, the vector velocity is tangent to the surface and the disturbance velocity vanishes as the distance from the airfoil increases to infinity.

When the flow is not symmetrical, we need an additional

condition which is given by the Kutta requirement. The Kutta condition states that the flow cannot go around the sharp trailing edge, but must leave the airfoil so that the upper and lower streams join smoothly at the trailing edge. This condition determines a unique value of the circulation.

There are many techniques for calculating the incompressible potential flow around two-dimensional bodies; this chapter reviews several methods which have some similarity to the current singular integral method.

1.2 Thin-Airfoil Theory

The thin-airfoil theory uses several approximations in order to calculate the surface pressure distribution. The method of calculation is convenient for a rapid estimation of the velocity or pressure distribution over the airfoil. This theory had its beginnings in the early days of Thermodynamics with Munk [1], Birnbaum [2] and Glauert [3].

We assume that the airfoil is thin and that the camber and angle of attack are small. This suggests that the disturbance velocity is small compared to the free steam velocity $\mathbf{U}_\infty$. Since at the stagnation point this statement is evidently not true, the calculation is useful in regions which exclude this point.

The differential equation governing the flow field is:

$$\nabla^2\phi = 0 \tag{1.4}$$

The boundary condition along the mean camber line given by,

$$\frac{\partial\phi}{\partial y} = U_\infty \frac{\partial\eta}{\partial x} - U_\infty \alpha \tag{1.5}$$

where η is the equation of the camber line, U_∞ is the free stream velocity and α is the angle-of-attack. The far stream condition can be stated as:

$$\frac{\partial\phi}{\partial x}, \frac{\partial\phi}{\partial y} \rightarrow 0 \quad \text{at infinity.} \tag{1.6}$$

The solution for the flow field is obtained by superposing three problems:

Problem 1 represents a thin symmetrical airfoil at zero angle of attack.

Problem 2 represents the steady flow past a cambered airfoil of zero thickness at zero angle of attack

Problem 3 represents a flat plate airfoil at an angle of attack.

We first consider the problem of a thin symmetrical airfoil at zero angle of attack. The effect of thickness can be represented by a continuous distribution of sources along the x-axis. The disturbance potential can then be expressed by the following integral equation:

$$\phi(x,y) = \frac{1}{2\pi}\int_0^1 q(\xi)\log\left[(x-\xi)^2 + y^2\right]^{\frac{1}{2}} d\xi \tag{1.7}$$

The proper source distribution denoted by $q(\xi)$ is determined by the surface boundary condition which gives:

$$q(x) = 2 U_\infty \frac{\partial \eta}{\partial x} \tag{1.8}$$

For the problem of the cambered airfoil of zero thickness we use a suitable distribution of vortices along the X-axis of the airfoil. The disturbance potential is given at the field point (x , y) by the integral relation:

$$\phi(x,y) = -\frac{1}{2\pi} \int_0^1 \gamma(\xi) \tan^{-1}\left[\frac{y}{(x-\xi)}\right] d\xi \tag{1.9}$$

Again the boundary conditions allow us to evaluate the axial distribution of vorticity.

The flow field over a flat plate airfoil is solved by using a vortex distribution. The suitable distribution of vorticity is given as a solution of the following integral equation:

$$\frac{1}{2\pi} \int_0^1 \gamma(\xi) \frac{d\xi}{x-\xi} = -U_\infty \alpha \tag{1.10}$$

After some mathematical manipulation, we obtain the vortex distribution:

$$\gamma(\theta) = 2 U_\infty \alpha \frac{1-\cos\theta}{\sin\theta} \tag{1.11}$$

The relationship between x and θ is given by:

$$x = \frac{1}{2}(1 + \cos\theta) \tag{1.12}$$

The general solution for the flow over a thin airfoil is finally obtain by superposing the three previous calculations. This developments come from reference [4]

1.3 Surface Distributions for Potential Flow

The principle of this method is to sum sources, sinks, vortices or dipoles on the surface of the airfoil to form a flow field that satisfies the boundary conditions. The surface of the airfoil is approximated by N elements or panels with N points at which the singularity is to be evaluated. Each singularity is superposed with the uniform free stream and the resulting flow velocity must be tangent to each N elements of the airfoil at the points where the singularity is to be evaluated. All surface singularity methods, sometimes called panel methods, use the zero normal velocity on the surface to derive an integral equation for the singularity distribution. The evaluation of the proper distribution basically solves the problem and allows the computation of the pressure distribution. The most straightforward form to formulate this method is to use Green's theorem. The potential at any point P exterior to the airfoil can be expressed as:

$$\phi(P) = -\frac{1}{4\pi}\iint_S \frac{1}{r(P,q)}\frac{\partial\phi}{\partial n_q}(q)\,ds + \frac{1}{4\pi}\iint_S \phi(q)\,\frac{\partial}{\partial n_q}\left[\frac{1}{r(P,q)}\right] ds$$

(1.13)

n denotes the normal to the surface at point q. The potential at a point p on the surface is given by:

$$\phi(p) = -\frac{1}{4\pi}\iint_S \frac{1}{r(p,q)}\frac{\partial\phi}{\partial n_q}(q)\,ds + \frac{1}{4\pi}\iint_S \phi(q)\,\frac{\partial}{\partial n_q}\left[\frac{1}{r(p,q)}\right] ds$$

(1.14)

Since $\partial\phi/\partial n_q$ is prescribed, this is an integral for $\phi(r)$. This equation represents a Fredholm integral equation, whose Kernel is given by:

$$K(p,q) = -\frac{\partial}{\partial n_q}\left[\frac{1}{r(p,q)}\right] \qquad (1.15)$$

However, the formulation that is more convenient is given by a surface distribution of unknown source strength

$$\phi(P) = \iint_S \frac{1}{r(P,q)}\,\sigma(q)\,ds \qquad (1.16)$$

This type of distribution gives a unique solution for the potential flow. Applying the boundary condition, we obtain the following integral equation:

$$2\pi\,\sigma(p) - \iint_S \frac{\partial}{\partial n_p}\left[\frac{1}{r(p,q)}\right]\sigma(p)\,ds = -\,n_p \cdot U_\infty \qquad (1.17)$$

This is a Fredholm integral equation of the second kind whose kernel is:

$$K(p,q) = -\frac{\partial}{\partial n_q}\left[\frac{1}{r(p,q)}\right] \tag{1.18}$$

These equations are the basic formulation of the surface source density method to solve the problem of potential flow. The accuracy of this method is determined by the number of the elements used to approximate the body surface. For usual shapes, such as a two dimensional airfoil, 30 to 60 elements is sufficient. Usually the only interest is in the calculation of the surface velocity. The potential off the airfoil needs not to be calculated.

Lift is deduced by means of a vorticity distribution on the surface. A conventional airfoil has a sharp trailing edge therefore for each angle of attack, there is a unique circulation that makes the potential flow velocity finite at the trailing edge. This condition is known as the Kutta condition. One technique is to put a vortex surface distribution on the surface of the airfoil. This method was proven to give the most accurate solution. The votex distribution would take the same N elements used for the source distribution and all the components will be summed up over the elements. The variation of the strength of the vortices is arbitrary, however a constant strength gives the most accurate solution. Another

technique is to place the vorticity distribution on the airfoil mean line. In this case only one vortex singularity needs to be placed on the mean line. This formulation has been reviewed by Maskew and Woodward [5]

This method of solving the potential flow using a surface distribution is general and can be applied to any kind of bodies (two- dimensional, axisymmetric and even three dimensional shapes). Because of its versatility and accuracy, it has become the most popular technique for computing potential flows. This developments come from references [6] and [7].

1.4 Conformal Transformations

Conformal transformations solve the potential flow field by using a complex transformed plane. The method simplifies the calculation of the flow field by solving for the flow field over a circle in the transformed plane which corresponds to the flow over a complicated airfoil shape in the real plane. Laplace's equation in the real plane transforms into Laplace's equation in the virtual plane and also the boundary conditions remain the same in both planes. The transformation maps points from the real plane using complex variables:

$$Z = X + iY \tag{1.19}$$

into points on a transformed plane:

$$\zeta = \xi + i\eta \tag{1.20}$$

The mapping is given by a function of the type:

$$Z = \zeta + r_1/\zeta + r_2/\zeta^2 + r_3/\zeta^3 + \ldots \tag{1.21}$$

where r_i are real constants. The problem in the transformed plane reduces to the flow field over a cylinder. The exact solution is found by using a doublet, a uniform free stream and a value for the circulation which ensures the uniqueness of the solution. The solution is then transformed back into the real plane to give the pressure distribution on the airfoil. This method is limited to special airfoil profiles for which a conformal transformation exists.

The most important conformal transformation is the Joukowsky transformation which leads to a family of airfoils known as Joukowsky airfoils. The Joukowsky transformation has the form:

$$Z = \zeta + r/\zeta \tag{1.22}$$

and the inverse transformation used to transform back the solution into the real plane is given by:

$$\zeta = \frac{Z}{2} \pm \left[\frac{Z^2}{4} - r^2 \right]^{\frac{1}{2}} \tag{1.23}$$

In the transformed plane the center of the circle is displaced from the origin and the X-axis displacement is proportional to the thickness of the Joukowsky airfoil while the Y-axis

displacement is determined by the camber. There are an infinite number of flows over a circle and the unique solution is found by invoking the Kutta condition which determined the position of the rear stagnation point on the circle. Another important aspect of the transformation is that at infinity the flows have exactly the same form and the angle of attack is also the same in either plane. More details about the Joukowsky transformation can be found in the reference [8]. Real airfoils are not Joukowsky airfoils, however the study of Joukowsky airfoil shapes give general trends for ideal flow over real airfoil shapes of similar thickness and camber.

CHAPTER 2
MATHEMATICAL FORMULATION AND NUMERICAL ALGORITHM

2.1 Introduction

The problem of predicting the pressure distribution about a two-dimensional airfoil has received considerable attention from various investigators. This study uses a recently developed [9] singular integral technique to solve the potential flow field over a two-dimensional airfoil. The calculation technique bears some resemblance to conventional singular integral methods as it reduces the formulation of the two-dimensional flow problem to the solution of an integral equation. However, this method has significant differences from other methods. It derives the integral equation by using a Fourier transform. Then, introducing a Taylor series expansion, the inverse transform is evaluated analytically. An important aspect is that the surface geometry of the airfoil appears explicitly in the integral equation. The advantage of this technique is that the formulation allows the inverse calculation to be easily performed, i.e. given a desired pressure distribution, the airfoil geometry can be found without iteration.

Two main computer programs have been developed for

two-dimensional airfoils. The first program will compute a pressure distribution for arbitrary combinations of airfoil geometry and angle of attack while the second program will calculate an airfoil profile for a given pressure distribution. The programs were shown to be in good agreement with known results. The numerical results will be presented in a subsequent chapter. This chapter descibes the mathematical development and numerical solution associated with this new formulation of the potential flow problem.

2.2 Formulation of the Equations

Consider a symmetrical airfoil of moderate thickness at some angle of attack to a free stream. The outer flow is two-dimensional, inviscid and irrotational. As shown in Figure 2.1, the cartesian coordinate system is taken with the origin at the center point of the airfoil chord. The length of the airfoil chord is taken using non-dimensional variables to be equal to 2. The outer free stream at infinity U_∞ is inclined at an angle of incidence α relative to the x-axis. The equation of the airfoil relative to the axis system is denoted by:

$$y=\eta(x) \qquad (2.1)$$

We introduce a disturbance velocity vector $q(x,y)$ due to the presence of the airfoil and we write:

$$\vec{U} = \vec{U}_{\infty} + \vec{q}\,(x,y) \tag{2.2}$$

In the following analysis, U_{∞} represents a constant vector. In non-dimensional form, the magnitude of U_{∞} is equal to 1. In a similar way, we introduce a disturbance potential for the disturbance velocity and we have:

$$\phi = \phi_{\infty} + \tilde{\phi} \tag{2.3}$$

It should be noted that the disturbance potential does not need to be small in the following formulation.

The first step in the formulation is to divide the problem into an upper and lower plane as shown in Figure 2.2. The mathematical problem will then be solved independently for the lower and upper plane. In terms of the potential the flow field is described by the following mathematical formulation for the upper plane:

Differential equation

$$\frac{\partial^2 \phi^u}{\partial x^2} + \frac{\partial^2 \phi^u}{\partial y^2} = 0 \tag{2.4}$$

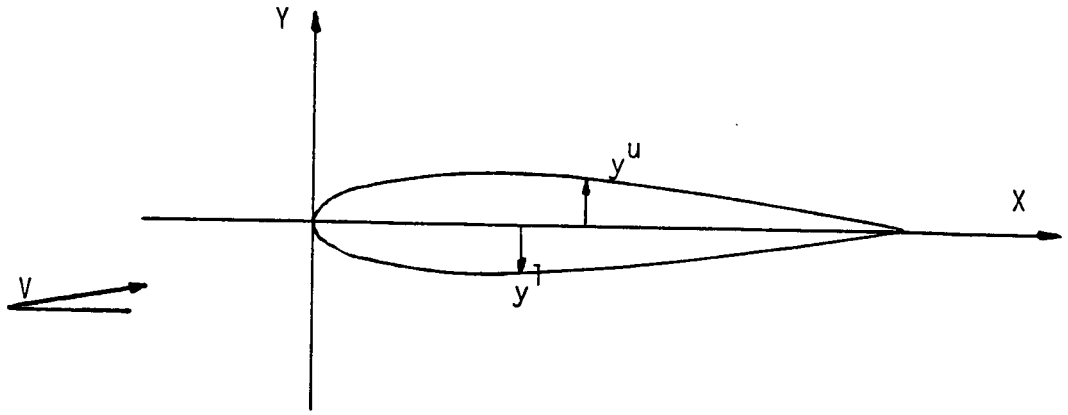


Figure 2.1

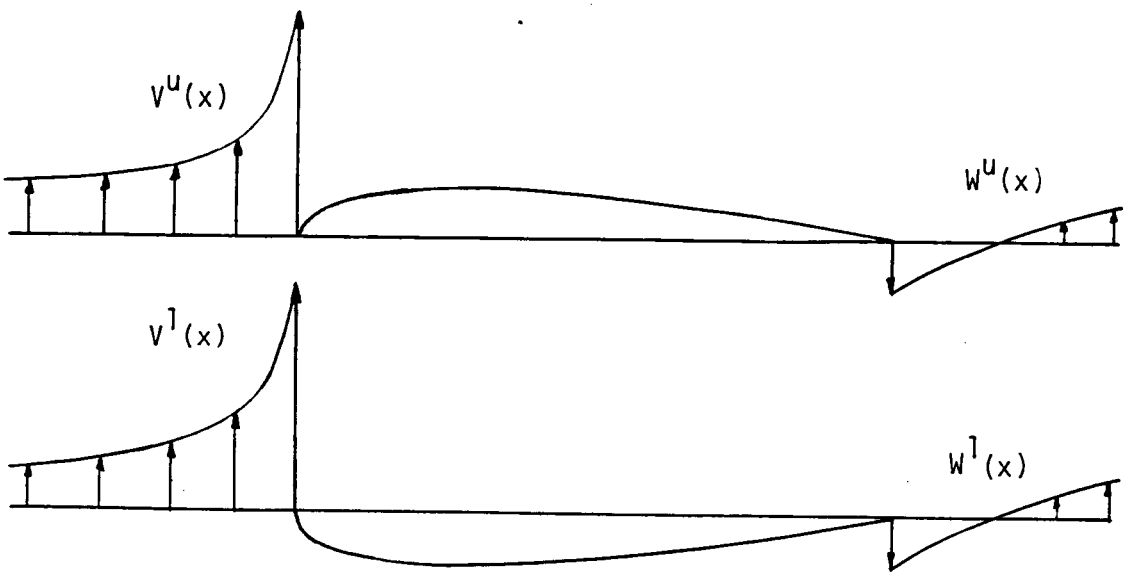


Figure 2.2

Surface boundary condition

$$\left[\frac{\partial \phi^u}{\partial y}\right]_{y=\eta(x)} = \begin{cases} v(x) & -\infty < x < -1 \\ \dfrac{\partial \phi^u}{\partial x}\dfrac{\partial \eta^u}{\partial x} & -1 < x < 1 \\ w(x) & 1 < x < \infty \end{cases} \qquad (2.5)$$

Far stream condition

ϕ^u (x,y) equal ϕ_∞ as the distance from the airfoil increases to infinity

For the lower plane a similar set of equations applies and the potential function will be denoted by $\phi^l(x,y)$. For symmetrical flows with a symmetrical airfoil only the upper half needs to be computed. However, in the general case, both the upper and lower plane are independently calculated. The parameters, $v^u(x)$ and $w^u(x)$ are the y-components of the velocities along the centerline upstream and downstream of the airfoil respectively. For a symmetrical flow $v^u(x)$ and $w^u(x)$ are zero. For nonsymmetrical flows, we require that $(v^u(x) , w^u(x))$ and $(v^l(x) , w^l(x))$ match along the centerline, this ensures a unique solution and in effect replaces the usual Kutta condition.

Finally we rewrite the mathematical formulation in terms of the disturbance potential and the disturbance upwash and downwash velocities defined by:

$$U(x) = \sin\alpha + \tilde{u}(x)$$
$$W(x) = \sin\alpha + \tilde{w}(x) \qquad (2.6)$$

The mathematical statement of the disturbance potential is given below for the upper plane:

$$\frac{\partial^2 \tilde{\phi}^u}{\partial x^2} + \frac{\partial^2 \tilde{\phi}^u}{\partial y^2} = 0 \qquad (2.7)$$

$$\left[\frac{\partial \tilde{\phi}^u}{\partial y}\right]_{y=\eta(x)} = f^u(x) \qquad (2.8)$$

where,

$$f^u(x) = \begin{cases} \tilde{U}(x) & -\infty < x < -1 \\ \left(\cos\alpha + \frac{\partial \tilde{\phi}^u}{\partial x}\right)\frac{\partial \eta^u}{\partial x} - \sin\alpha & -1 \le x \le 1 \\ \tilde{W}(x) & 1 < x < \infty \end{cases}$$

$$\frac{\partial \tilde{\phi}^u}{\partial x} \text{ and } \frac{\partial \tilde{\phi}^u}{\partial y} \to 0 \quad \text{as} \quad x^2 + y^2 \to \infty \qquad (2.9)$$

A similar set of equations exists for the lower plane.

2.3 Solution by Fourier Transform

In order to find a solution for the above mathematical equations we use the Fourier Transform in the x-direction assuming that the function ϕ satisfies the Dirichlet conditions

in every finite interval. The Fourier transform pair is given by:

$$\Phi(s,y) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} \tilde{\phi}(x,y)\, e^{-isx}\, dx \tag{2.10}$$

$$\tilde{\phi}(x,y) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} \Phi(s,y)\, e^{isx}\, ds \tag{2.11}$$

The function $\Phi(s,y)$ is the Fourier transform of $\phi(x,y)$. The differential equation for $\Phi(s,y)$ is given by,

$$\frac{\partial^2 \Phi}{\partial y^2} - s^2 \Phi = 0 \tag{2.12}$$

with the boundary conditions:

$$\left[\frac{\partial \Phi}{\partial y} \right]_{y=\eta(s)} = F(s) \tag{2.13}$$

$$\Phi \text{ and } \frac{\partial \Phi}{\partial y} \to 0 \quad \text{as} \quad y \to \infty$$

$$F(s) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} f(x)\, e^{-isx}\, dx \tag{2.14}$$

$$f(x) = \begin{cases} \tilde{U}(x) & -\infty < x < -1 \\ \left(\cos\alpha + \dfrac{\partial \tilde{\phi}}{\partial x} \right) \dfrac{\partial \eta}{\partial x} - \sin\alpha & -1 \le x \le 1 \\ \tilde{W}(x) & 1 < x < \infty \end{cases} \tag{2.15}$$

The solution is easily found to be:

$$\Phi(s,y) = -\frac{F(s)}{|s|} e^{-|s|(y-\eta(s))} \tag{2.16}$$

Applying the inversion formula , the potential function is formally given by:

$$\tilde{\phi} = -\frac{1}{2\pi}\int_{-\infty}^{\infty} f(\xi)\left[\int_{-\infty}^{\infty} \frac{e^{-|s|(y-\eta(s))}\, e^{is(x-\xi)}}{|s|}\, ds\right] d\xi \tag{2.17}$$

Although the solution is exact, the above equation is not useful from a computational perspective. However a simplified expression is found by expanding the exponential in a Taylor series over the transform variable **s**. A term by term integration then yields the following approximate expressions:

$$\frac{\partial\tilde{\phi}}{\partial x} = \frac{1}{\pi}\int_{-\infty}^{\infty} f(\xi)\, K_x(x,y;\xi)\, d\xi$$
$$\frac{\partial\tilde{\phi}}{\partial y} = \frac{1}{\pi}\int_{-\infty}^{\infty} f(\xi)\, K_y(x,y;\xi)\, d\xi \tag{2.18}$$

The Kernel functions of the above integral equations, accurate to $O(\varepsilon^2)$ where $\varepsilon = t/c$, are given by:

$$K_x(x,y;\xi) = \frac{x-\xi}{(x-\xi)^2+(y-\eta)^2}$$
$$K_y(x,y;\xi) = \frac{y-\eta}{(x-\xi)^2+(y-\eta)^2} \tag{2.19}$$

Although, (2.18) and (2.19) represent approximate solutions in general, it can be shown that they are exact on the surface. Also, it should be noted that if $\eta=0$, and if the disturbance potential is set to zero in equation (2.15), we recover the usual thin airfoil equations.

It may be observed that the equations (2.18) are singular Fredholm integral equations of the second kind and their Kernels are of difference type with a Cauchy singularity. For example, the behavior of K_x is: $K_x(x,y;\xi) \rightarrow \pm\infty$, as $\xi \rightarrow x_0$. The variation of $f(\xi)$ in the Kernel function is continuous across the interval [-1,1] exept at the leading edge. The treatement of the singularity condition at $\xi=x$ requires the use of the Cauchy principal values. Another important aspect is that the Fredholm integral equation must be solved in an iterative manner since $f(\xi)$ contains a term $\partial\phi/\partial x$ which is unknown. The solution of the equations (2.18) by means of various optimized quadrature techniques is discussed in the following section.

2.4 Outline of the Singular Integral Equation Solution Procedure

We wish to solve the integral equation (2.18) on the surface of the airfoil with $y = \eta(x)$. First we replace the

function $f(x)$ by its respective values along the x-axis upstream and downstream of the airfoil and on the surface of the airfoil. The resulting integral equation then becomes:

$$\frac{\partial \tilde{\phi}^u}{\partial x} = \frac{1}{\pi}\int_{-\infty}^{-1} \tilde{U}(\xi)\,\frac{d\xi}{x-\xi} + \frac{1}{\pi}\int_{1}^{\infty} \tilde{W}(\xi)\,\frac{d\xi}{x-\xi} + \frac{1}{\pi}\int_{-1}^{1}\left[\left(\cos\alpha + \frac{\partial \tilde{\phi}^u}{\partial \xi}\right)\frac{d\eta^u}{d\xi} - \sin\alpha\right]\frac{d\xi}{x-\xi} \tag{2.20}$$

The general procedure used in this calculation method is to represent the prescribed function $f(x)$ along the airfoil surface by a linear combination of basis functions, in this case, powers of x:

$$f(x) = \sum_{i=0}^{n} a_i\, x^i \qquad -1 < x < 1 \tag{2.21}$$

or,

$$f(x) = \sum_{i=0}^{n} (1-x^2)\, a_i\, x^i \qquad -1 < x < 1 \tag{2.22}$$

Both functions have been tested, along with several other basis functions. The polynomial (2.21) yields better results and the remainder of this chapter will be devoted to computation methods using the polynomial (2.21).

The coefficients a_i are unknown and the function $f(x)$ is a polynomial of degree n. We solve for the coefficients a_i from

equation (2.20). These values will depend on the choice of the quadrature points x_j and the degree of the power series. The integral operand in the airfoil integral equation is not a continuous function on [-1, 1] and the equation (2.20) is a singular integral equation which can be solved using the Cauchy principal values on the interval [-1, 1] . Replacing $f(x)$ by a power series, equation (2.20) may be written as:

$$\frac{\partial \tilde{\phi} u}{\partial x} = \frac{1}{\pi}\int_{-\infty}^{-1} \tilde{U}(\xi)\,\frac{d\xi}{x-\xi} + \frac{1}{\pi}\int_{1}^{\infty} \tilde{W}(\xi)\,\frac{d\xi}{x-\xi} + \frac{1}{\pi}\int_{-1}^{1}\left[\sum_{i=0}^{n} a_i\,\xi^i\right]\frac{d\xi}{x-\xi} \qquad (2.23)$$

The last integral of the right hand side can be conveniently evaluated term by term by using the Cauchy principal values given in the appendix:

$$a_0\int_{-1}^{1}\frac{d\xi}{x-\xi} = a_0\,\ln\left(\frac{1+x}{1-x}\right)$$

$$a_1\int_{-1}^{1}\frac{\xi\,d\xi}{x-\xi} = a_1\left[\,x\,\ln\left(\frac{1+x}{1-x}\right) - 2\,\right] \qquad (2.24)$$

Finally any term of degree i can be written as a function of the term of degree i-1 :

$$\int_{-1}^{1} \frac{\xi^i}{x-\xi} d\xi = x \int_{-1}^{1} \frac{\xi^{i-1}}{x-\xi} d\xi - \frac{1-(-1)^i}{i} \qquad (2.25)$$

In equation (2.23), the coefficients a_i are unknown and we define a set of n+1 collocation points x_j as shown in figure 2.3, where the integral equation (2.23) is to be evaluated at each point. The basic integral (2.23) can be expressed at any point x_j as a linear combination of the coefficients, a_i . This coefficients are obtained by numerical integration and are a function of the geometry of the airfoil. Application of the above conditions gives a set of n+1 linear equations for the n+1 unknown values of a_i.

2.4.1 Numerical Solution for Symmetrical Flow Field

Consider first the case of a symmetrical airfoil at zero angle of attack. We see that the disturbance velocities $U(x)$ and $W(x)$ are null. Inspection of (2.23) yields the expression:

$$\frac{\partial \tilde{\phi}}{\partial x} u = \frac{1}{\pi} \left[\sum_{i=0}^{n} a_i I_i \right] \qquad (2.26)$$

where I_i is the Cauchy integral of degree i, obtained from the integration of the basis functions. If we use the basis functions given by equation (2.21), and note that:

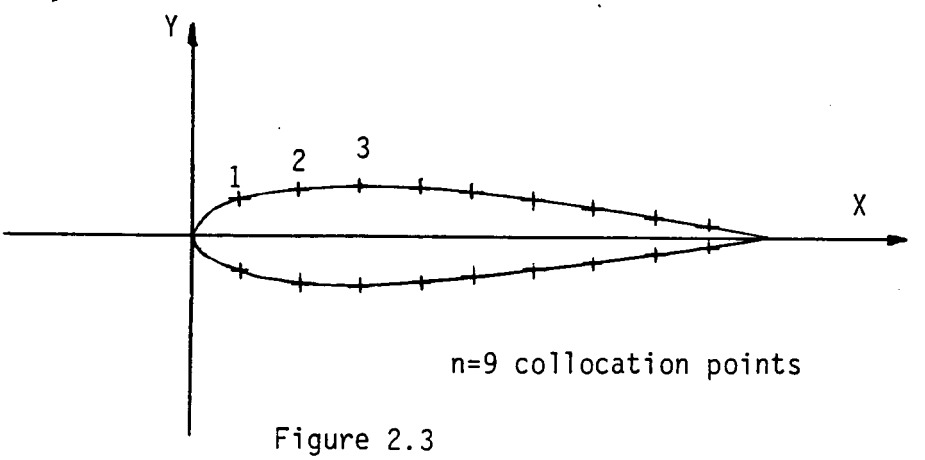


Figure 2.3

$$f(x) = \left(1 + \frac{\partial \tilde{\phi}^{u}}{\partial x}\right) \frac{d\eta}{dx} \tag{2.27}$$

we obtain:

$$\left(1 + \frac{\partial \tilde{\phi}^{u}}{\partial x}\right) \frac{d\eta}{dx} = \sum_{i=0}^{n} a_i x^i \tag{2.28}$$

Solving of $\phi_x{}^u$, and substituting into equation (2.26) yields:

$$\sum_{i=0}^{n} a_i \left[\frac{I_i}{\pi} - \frac{x^i}{\frac{\partial \eta}{\partial x_j}} \right] = -1 \tag{2.29}$$

This equation applies at any x_j location between -1 and +1. Applying this equation at n+1 points yields n+1 equations for the n+1 unknowns.The coefficients of a_i are called m_{ij} and define a matrix of dimension n+1. The solution to the airfoil equations is finally obtained by inverting the matrix m_{ij} by Gaussian triangularization and the coefficients of the function f(x) are given by:

$$a_i = m_{ij}{}^{-1} \cdot (-1)_j$$

The computational time to solve the matrix by triangularization is less than 5 seconds using the CDC Dual Cyber System for a matrix of dimension no larger than 40. The computing time is independent of the geometry of the airfoil,

however it is dependent of the number of collocation points **n**. In order to reduce the round-off errors, we compute the matrix with double precision variables with 16 decimal accuracy which is useful for a matrix of dimension higher than 30 since the determinant of the matrix is a small number. The solution breaks down for **n**≥40. This difficulty is caused by small valuesin the determinant when **n** is large. For high values of n, quantities of similar values are substracted in the calculation of the determinant which results in a dangerous loss of accuracy in the value of the determinant. consequently, the accuracy decreases as the number of quadrature points increases. On the other hand, using a small number of points may not be sufficient to define the shape of the body especially very near the leading or stagnation point. However, for sufficiently small n the use of single precision variables allows a quicker computing time and the single precision calculation gives satisfactory results. Finally the pressure distribution is obtained by substituting the values a_i into the following expression:

$$Cp = 1 - \left[f(x)\right]^2 \left[1 + \frac{1}{\left(\frac{\partial \eta}{\partial x}\right)^2}\right] \tag{2.30}$$

$$\text{with } f(x) = \sum_{i=1}^{n} a_i x^{i-1} \tag{2.31}$$

2.4.2 Numerical Procedure for Non-Symmetrical Flow Field

In addition to the solution of the basic potential flow problem over a symmetrical airfoil at zero angle of attack, the solution to the non-symmetrical flow field has been incorporated into the numerical method. For this case, the upwash $U(x)$ and downwash $W(x)$ are unknown. Consequently, both the upper and lower planes must be solved simultaneously and the solutions must be matched along the cut. A possible methodology is to use some initial guess for $U(x)$ and $W(x)$ and iterate until the change in the upwash and downwash is sufficiently small. The set of equations for this methodology are given below:

$$\frac{\partial \tilde{\phi}^U}{\partial x} = \frac{1}{\pi}\int_{-\infty}^{-1} \tilde{U}(\xi)\,\frac{d\xi}{x-\xi} + \frac{1}{\pi}\int_{1}^{\infty} \tilde{W}(\xi)\,\frac{d\xi}{x-\xi} + \frac{1}{\pi}\int_{-1}^{1}\left[\left(\cos\alpha + \frac{\partial \tilde{\phi}^U}{\partial \xi}\right)\frac{d\eta^U}{d\xi} - \sin\alpha\right]\frac{d\xi}{x-\xi} \tag{2.32}$$

$$\frac{\partial \tilde{\phi}^L}{\partial x} = \frac{-1}{\pi}\int_{-\infty}^{-1} \tilde{U}(\xi)\,\frac{d\xi}{x-\xi} - \frac{1}{\pi}\int_{1}^{\infty} \tilde{W}(\xi)\,\frac{d\xi}{x-\xi} - \frac{1}{\pi}\int_{-1}^{1}\left[\left(\cos\alpha + \frac{\partial \tilde{\phi}^L}{\partial \xi}\right)\frac{d\eta^L}{d\xi} - \sin\alpha\right]\frac{d\xi}{x-\xi} \tag{2.33}$$

An alternative approach, and one which proves to be superior, is to use an approximate representation for $U(x)$ and

$W(x)$ and avoid the iteration procedure. This is accomplished by representing the actual airfoil by an "equivalent" Joukowsky airfoil of the same thickness for the purpose of obtaining $U(x)$ and $W(x)$ only. This procedure yields an approximate representation for the upwash and downwash. An extensive numerical investigation showed that the solution to (2.32) and (2.33) is sufficiently insensitive to this approximation to justify its application. A lengthy analysis is necessary to describe the flow about a Joukowsky airfoil and the details of the calculation are given in appendix A. The result for the upwash and downwash disturbance velocities are given by the following expressions:

$$U(x)=\sqrt{\frac{x-1}{x+1}}\left[\left[\frac{\frac{x}{2}+\sqrt{\frac{x^2}{4}-\frac{1}{4}}}{\frac{x}{2}+\sqrt{\frac{x^2}{4}-\frac{1}{4}}+m}\right]^2-1\right]\sin\alpha \qquad (2.34)$$

$$W(x)=\sqrt{\frac{x-1}{x+1}}\left[\left[\frac{\frac{x}{2}-\sqrt{\frac{x^2}{4}-\frac{1}{4}}}{\frac{x}{2}-\sqrt{\frac{x^2}{4}-\frac{1}{4}}+m}\right]^2-1\right]\sin\alpha \qquad (2.35)$$

It should also be remembered that in the Joukowsky calculation the position of the leading edge is slighty different than -1 , for example it is equal to -1.014405 for a thickness of 12%.

The upwash and downwash integral terms of equations (2.32) and (2.33) are solved numerically by a simple trapezoidal rule. Boundary conditions on the airfoil surface are applied and the matrix elements are calculated using the same set of points distributed on the airfoil surface as for the symmetrical flow. Although there will be some additional terms in the calculation of the matrix elements, the numerical procedure remains the same. The matrix elements may be written as:

$$a_i\left[\frac{I_{ij}}{\pi}-\frac{\kappa_i^{i-1}}{\frac{\partial\eta}{\partial\kappa_j}}\right]=-\frac{1}{\pi}\int_{-\infty}^{-1}\tilde{U}^{u}(\xi)\frac{d\xi}{\kappa_j-\xi}-\frac{1}{\pi}\int_{1}^{\infty}\tilde{W}^{u}(\xi)\frac{d\xi}{\kappa_j-\xi}$$

$$+\sin\alpha\,\ln\left[\frac{1+\kappa_j}{1-\kappa_j}\right]-\cos\alpha \qquad (2.36)$$

A similar set of matrix elements applies for the lower plane.

2.5 Approximation of the Surface Boundary by n Quadrature Points

The integral equation described in the previous section is to be evaluated at a set of points κ_j distributed on the airfoil surface as shown in figure 2.3. Special attention should be taken when choosing the quadrature points since the accuracy of the calculation is fully determined by the number and the

distribution of the set of points. The quadrature of order **n+1** determines the number of unknown coefficients of the previously described function **f(x).**

The spacing of the points must be small compared to the dimensions of the airfoil. In addition the local curvature of the airfoil should be considered in the point distribution. The proper distribution of the points over the airfoil surface will be largely a matter of experience and intuituion. As a first approach and one that proves to give satisfactory results, we use a set of equally spaced points along the x-axis of the airfoil. The first point is located at a distance **d** from the leading edge with succeding points spaced the same distance **d.** However two serious problems arise from the sharp corner at the trailing edge and from the large slope at the leading edge. These areas need to be defined by using a higher concentration of points. A higher order implementation which uses parabolically varying distances between points has been applied to the airfoil problem. A high concentration of points occurs at the leading and trailing edge and varies toward the central region of the airfoil where the distribution is sparce. However, for high order implementations, longer computing times will be required and loss of accuracy may occur from round-off errors in the matrix calculation.

2.6 Solution of the Inverse Design Problem

2.6.1 Introduction

This section discusses an attempt to design by analytic means a class of airfoils using a similar methodology as for the direct problem. The design and development of aerodynamic bodies is usually an empirical procedure, based primarily upon the designer's experience and employing trial and error techniques. For the design problem, analytic solutions are not as developed as for the direct problem since it is more difficult and it involves the solution of a free boundary value problem. However, it is of great importance since for a desirable pressure distribution we can obtain the corresponding body shape.

In the literature, solutions to the design problem are mostly based on iteration techniques due to the absence of exact mathematical solutions for free boundary value problems. Marshall [10] presented a technique that removes the free boundary element by a perturbation procedure. An analytical solution, using a surface source distribution, is obtained in the form of integral equations. Nevertheless, the calculation method uses an initial guess and involves an iteration procedure. Zedan and Dalton [11] presented a method which employs an axial source-sink distribution, with constant element strength, to obtain a solution to the design problem.

The method proves to be accurate and converges, but it uses an iteration procedure and the method is also limited to bodies that do not present a sudden change in the slope of the meridian line. This present study does not requires any iteration and even less computational time is necessary than for the direct calculation.

2.6.2 Mathematical Formulation

In this section, the basic equations for the design problem with uniform flow field are derived. In this study, the method uses the surface velocity instead of the pressure as the prescribed distribution. The design problem can be stated as: given a surface velocity, what is the body shape that would produce this velocity distribution?

Again consider an inviscid incompressible flow over a two-dimensional airfoil. Equation (2.20) of the previous section remains applicable since the flow conditions remain the same: This equation is repeated for convenience.

$$\frac{\partial \tilde{\phi}^U}{\partial x} = \frac{1}{\pi}\int_{-\infty}^{-1} \tilde{U}(\xi)\,\frac{d\xi}{x-\xi} + \frac{1}{\pi}\int_{1}^{\infty} \tilde{W}(\xi)\,\frac{d\xi}{x-\xi} + \frac{1}{\pi}\int_{-1}^{1}\left[\left(\cos\alpha + \frac{\partial \tilde{\phi}^U}{\partial \xi}\right)\frac{d\eta^U}{d\xi} - \sin\alpha\right]\frac{d\xi}{x-\xi} \tag{2.37}$$

We are now faced with the problem of finding the shape

prescribed by $d\eta/d\xi$ given the surface velocity. As before, the function $f(x)$ may be expressed by a linear combination of powers of x.

$$f(x)=\sum_{i=0}^{n} a_i x^i \qquad -1 < x < 1 \tag{2.38}$$

where $f(x)$ is given by:

$$f(x) = \left(\cos\alpha + \frac{\partial \tilde{\phi}}{\partial x} \right)\frac{\partial \eta}{\partial x} - \sin\alpha = \frac{\partial \tilde{\phi}}{\partial y} \tag{2.39}$$

Equation (2.38) is evaluated at a set of $n+1$ quadrature points, which give a set of $n+1$ linear equations solved by Gaussian elimination, to obtain the $n+1$ values a_i. It should be noted that the x-disturbance velocity is actually the unknown at this point since the airfoil shape is still unknown. This component is determined by inserting the coefficients a_i in the integral equation (2.39). A similar procedure as the one used in the direct problem allows us to calculate the integrals of equation (2.37) by introducing the Cauchy principal values.

In order to gain better accuracy, the x-disturbance velocity is evaluated at 200 points along the chord length and the slope on the airfoil surface is ultimately given by the equation:

$$\frac{d\eta}{dx} = \frac{\frac{\partial \tilde{\phi}}{\partial y} + \sin\alpha}{\frac{\partial \tilde{\phi}}{\partial x} + \cos\alpha} \qquad -1 < x < 1 \qquad (2.40)$$

The treatment of the inverse design problem has been restricted to uniform flow at a zero degree of angle of attack which requires a less sophisticated approach since the location of the stagnation point is known.

Results of both the design and analysis problem determined by the procedure outlined above are presented in the next section.

CHAPTER 3

RESULTS FOR THE ANALYSIS MODE

3.1 Introduction

In this chapter, a series of numerical calculations for different airfoil geometries are presented. The results are validated by comparison to numerical solutions and analytical solutions when they exist. A code which was recently developed at NASA Langley [12] has been selected for purposes of verification of the present method. This code uses a spectral multigrid technique and has been extensively validated with finite difference schemes. In addition to this numerical verification, the present results are compared to analytical solutions for elliptical and Joukowski airfoils.

Data are presented in terms of the pressure coefficient C_p, which is the quantity of usual aerodynamic interest. It is defined, in general, as:

$$C_p = \frac{p - p_\infty}{\frac{1}{2} \rho U_\infty} \tag{3.1}$$

where p denotes the local pressure. In incompressible potential flow it is related to the velocity U by :

$$C_p = 1 - \left[\frac{U}{U_\infty} \right]^2 \tag{3.2}$$

The formulation of the problem, presented in Chapter 2, has been tested for the flow over a 12% thick elliptic airfoil, a NACA 0012 airfoil and a 12% thick Joukowski airfoil. A description of the results follows.

3.2 Implementation of the Quadrature Points

Flows have been computed using both equally spaced points and a higher order implementation. In the first method, the distribution of the points is simply determined by using a constant value Δ for the distance between two consecutive points. It should be also noted that the distance between the leading edge and the first collocation point as well as the distance between the last point and the trailing edge is equal to Δ.

The second method uses a geometrically increasing grid. The following equation is applied to determine the spacing between two consecutive points:

$$\delta x_j = \Delta \, r^j \qquad (3.3)$$

The resulting distribution is shown in Figure 3.1. Moving away from the leading edge, each one-dimensional grid spacing is made r times larger until the center of the airfoil is reached. The parameters Δ and r are constant values and j denotes the j^{th} interval. In order to evaluate the two unknown Δ and r, we

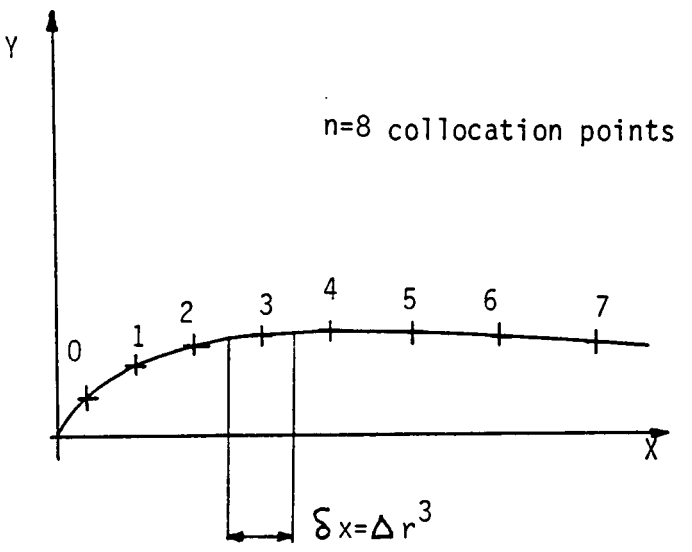


Figure 3.1

$$Cp = 1 - \frac{(1+\varepsilon)^2}{\left[1-\varepsilon^2\left[\frac{x^2}{1-x^2}\right]\right]} \qquad (3.6)$$

where ε denotes the thickness ratio.

Figure 3.2 and figure 3.3 compare the analytical solution with the calculated solution using 20 and 24 points distributed on the airfoil surface. Figure 3.2 uses the geometrically increasing grid and Figure 3.3 uses equally spaced points. The free stream velocity is parallel to the x-axis of the ellipse and as can be seen, the two plots are graphically indistinguishable for N=24. Positions of the points are shown in figures 3.4 and 3.5. Figure 3.2 and 3.3 are representative of several other calculations which were made using using a larger and smaller number of points and various types of grid points distributions.

3.4 NACA 0012 Airfoil

The airfoil profile is given by the equation:

$$y = 1.2(.2969\sqrt{x} - .12600x - .35160x^2 + .2840x^3 - .10150x^4) \qquad 1 \geq x \geq 0 \qquad (3.7)$$

Two calculated pressure distributions are shown for the NACA 0012 airfoil. One was calculated by using the methodology discussed in this study . The other distribution was obtained from a NASA computer code [8] and used as a comparison. As

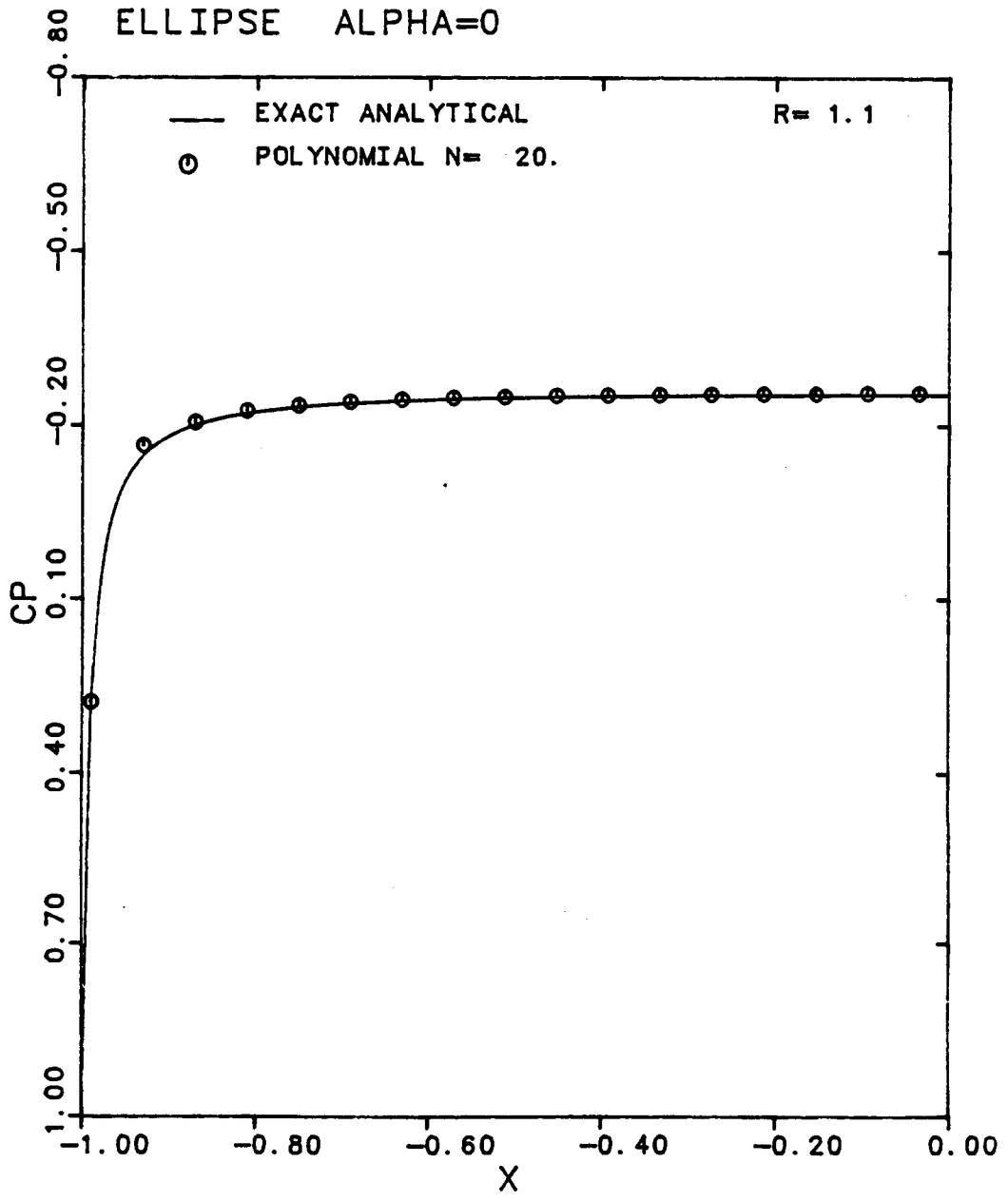

ELLIPSE ALPHA=0
EXACT ANALYTICAL
POLYNOMIAL N= 20.
R= 1.1
CP
X
-0.80
-0.50
-0.20
0.10
0.40
0.70
1.00
-1.00
-0.80
-0.60
-0.40
-0.20
0.00

Figure 3.2

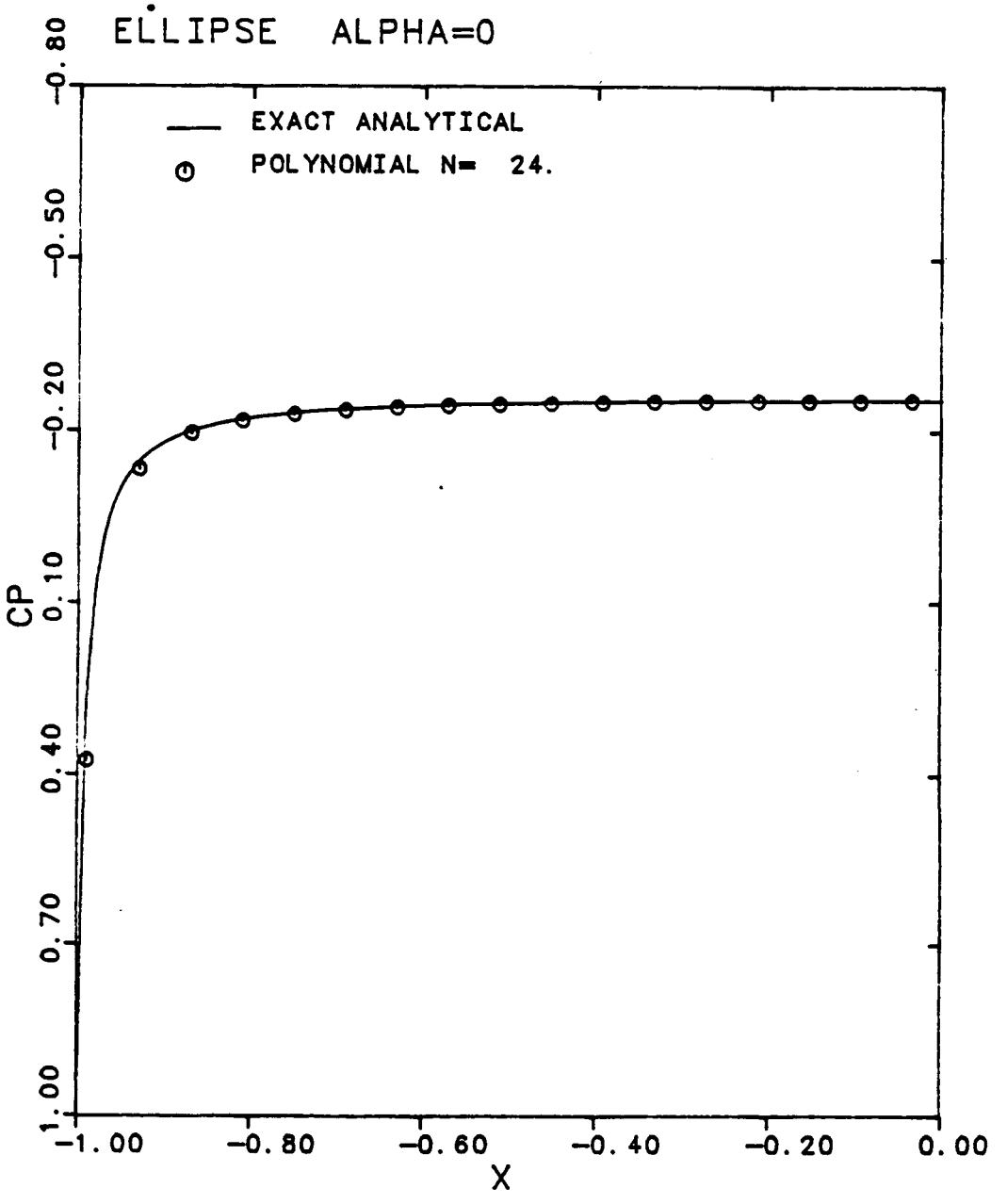

ELLIPSE ALPHA=0
EXACT ANALYTICAL
POLYNOMIAL N= 24.
CP
X
-0.80
-0.50
-0.20
0.10
0.40
0.70
1.00
-1.00
-0.80
-0.60
-0.40
-0.20
0.00

Figure 3.3

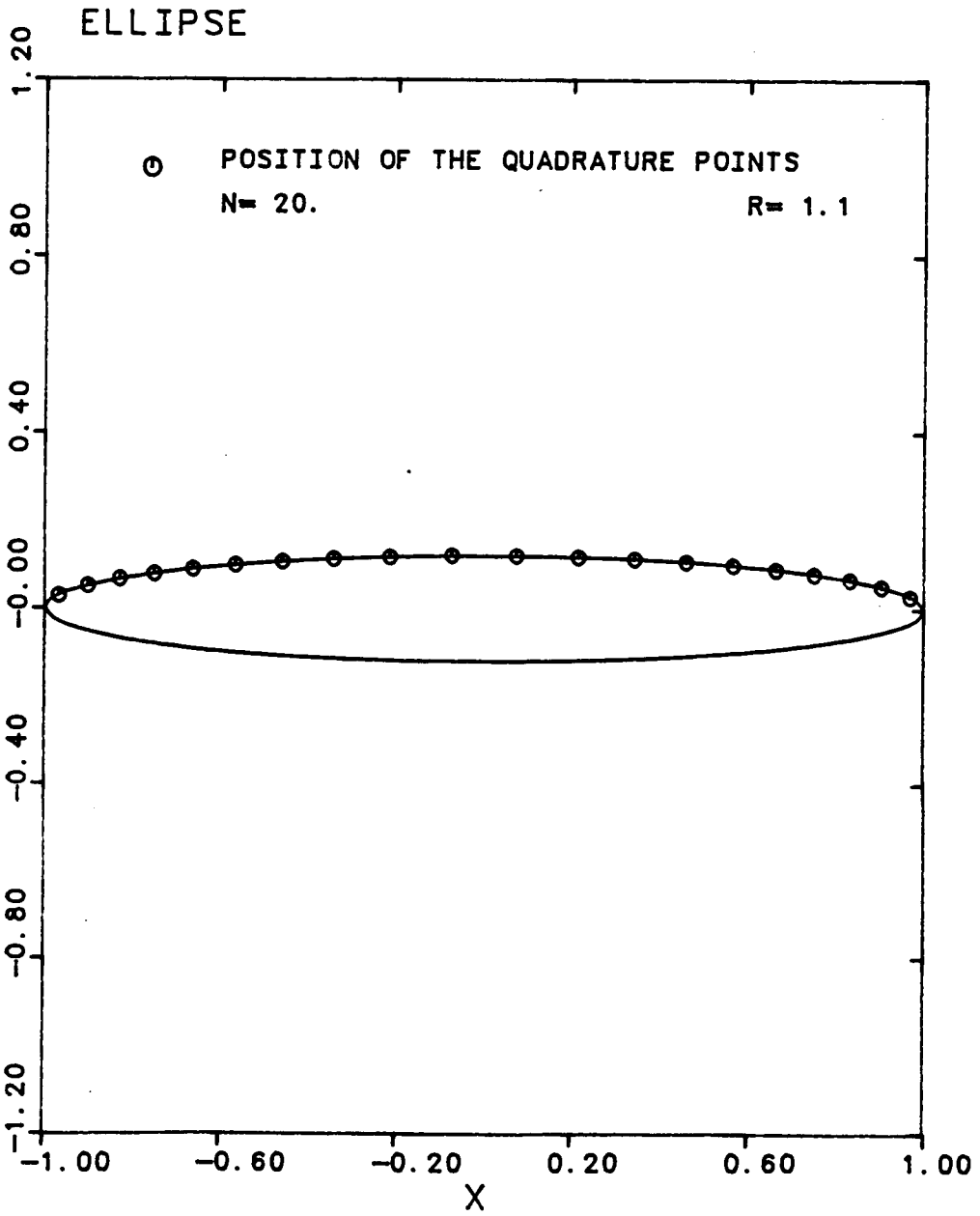

ELLIPSE
POSITION OF THE QUADRATURE POINTS
N= 20.
R= 1.1
1.20
0.80
0.40
-0.00
-0.40
-0.80
-1.20
-1.00
-0.60
-0.20
0.20
0.60
1.00
X

Figure 3.4

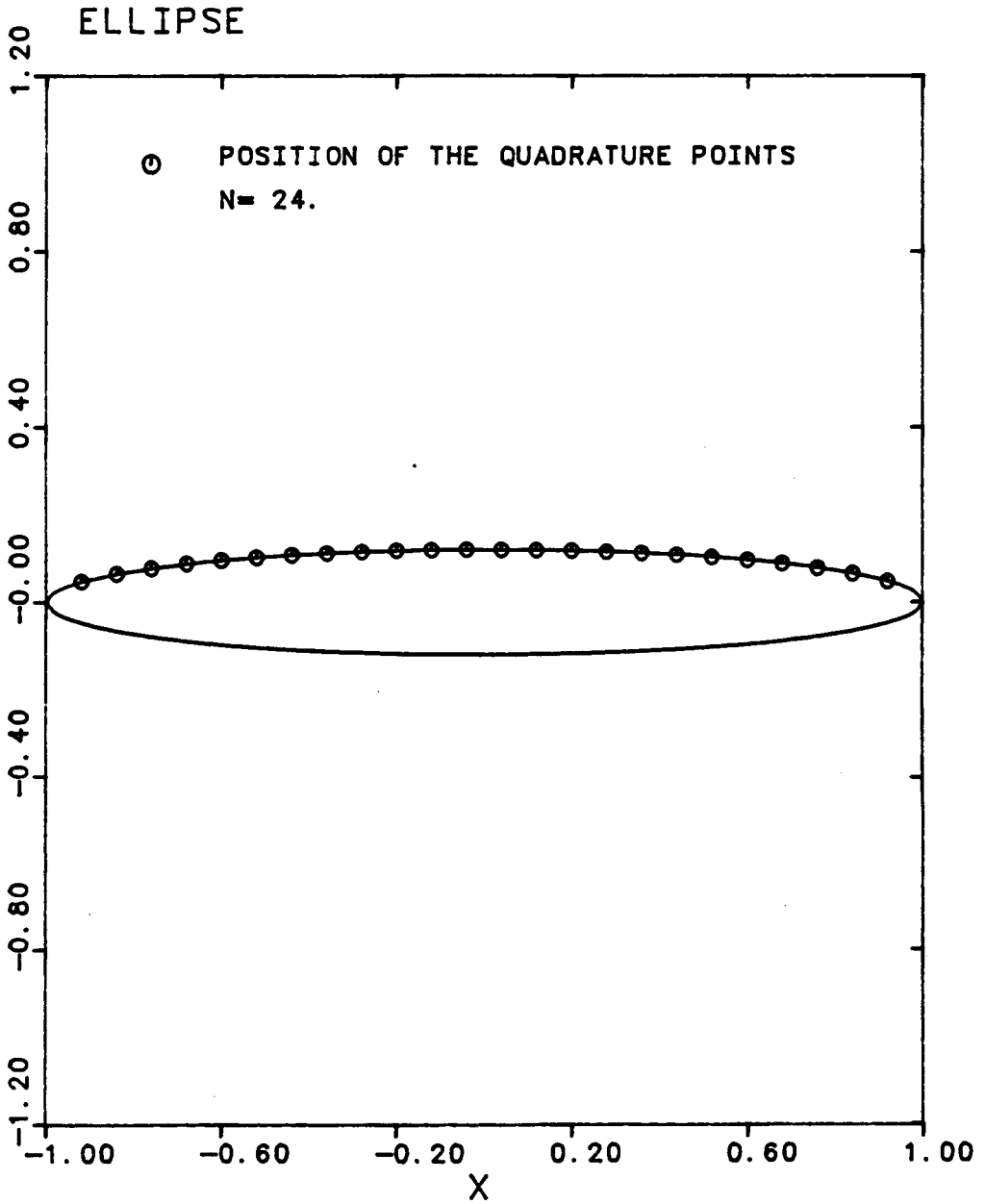

ELLIPSE
POSITION OF THE QUADRATURE POINTS
N= 24.
1.20
0.80
0.40
-0.00
-0.40
-0.80
-1.20
-1.00
-0.60
-0.20
0.20
0.60
1.00
X

Figure 3.5

illustrated in figures 3.6, 3.7 and 3.8, the two results are indistinguishable over the central region. Agreement with the NASA computer code can be improved at the trailing edge by adopting the variable grid scheme given by equation (3.3). This improvement can be seen by comparing figures 3.6 and 3.7 , but it should be noted that a slight loss of accuracy occurs in the region of the leading edge when using the geometrically increasing grid. Positions of the points are shown in figures 3.9 and 3.10 for the distributions of figures 3.7 and 3.8.

The calculations were repeated for different sets of quadrature points and the method has proven to give consistent results over a range of 15 to 30 points distributed over the airfoil surface. For N greater than approximatly 35, the accuracy begins to decrease due to the increasing matrix round-off error. A number of points smaller than 15 does not give an accurate description of the airfoil geometry. The calculated pressure coefficient exhibits a small repeated error very near the leading edge. This behavior can be explained by the difficulty that occurs when fitting a polynomial function over the region of large velocity gradients, e.g. the leading edge region. Slight changes in the locations and the number of the quadrature points can improve the accuracy of the curve.

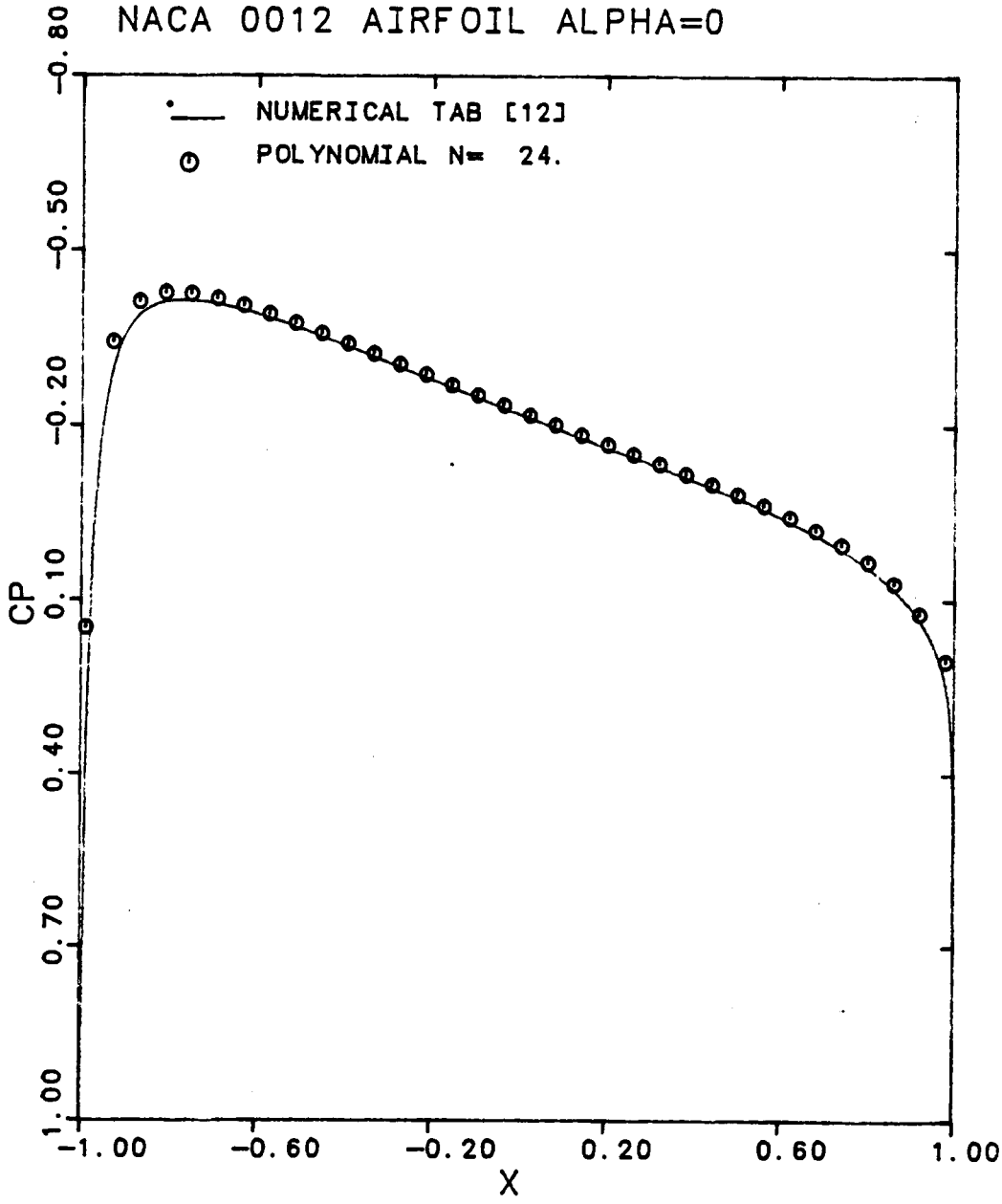

NACA 0012 AIRFOIL ALPHA=0
NUMERICAL TAB [12]
POLYNOMIAL N= 24.
CP
X
-0.80
-0.50
-0.20
0.10
0.40
0.70
1.00
-1.00
-0.60
-0.20
0.20
0.60
1.00

Figure 3.6

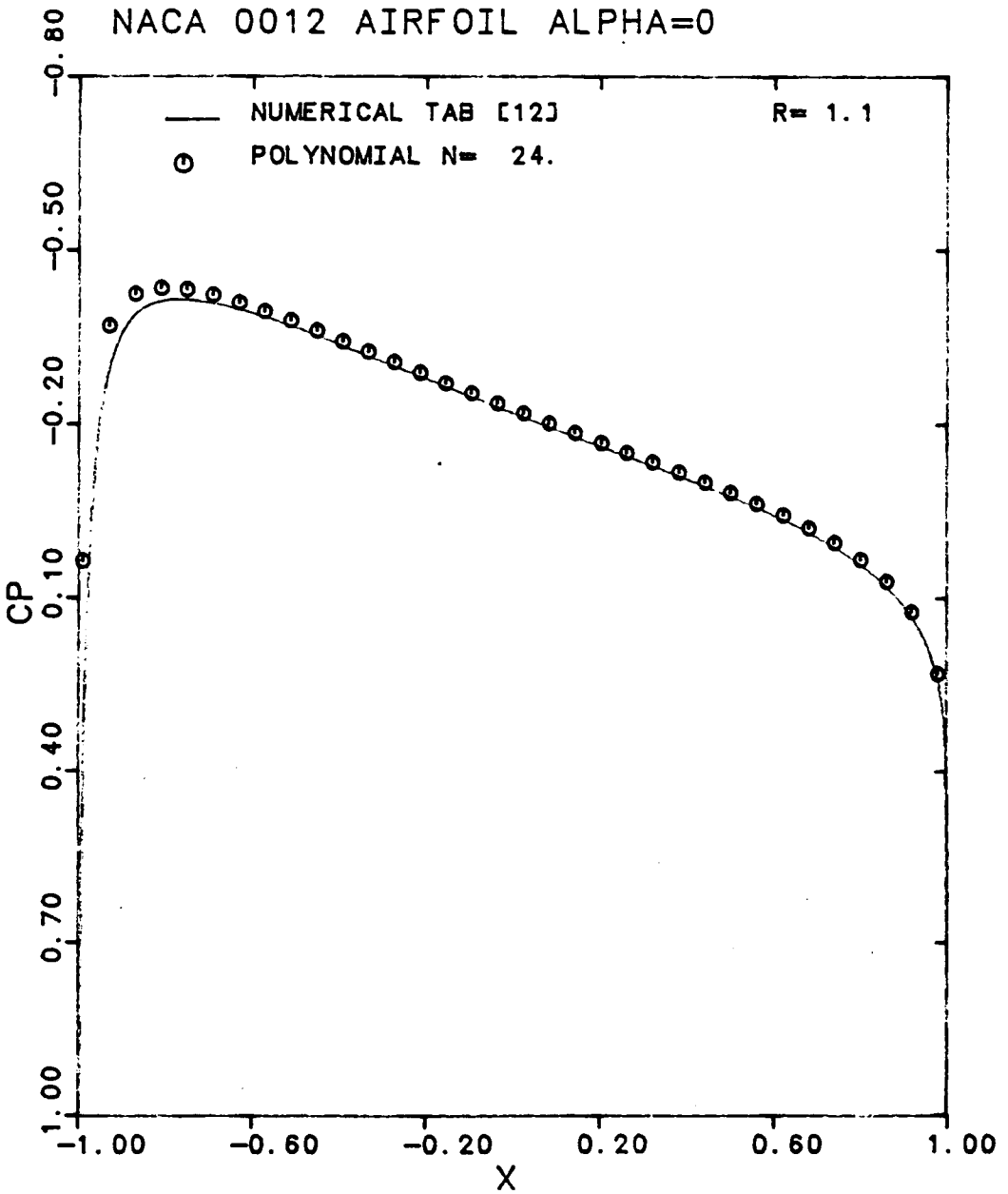

NACA 0012 AIRFOIL ALPHA=0
NUMERICAL TAB [12]
R= 1.1
POLYNOMIAL N= 24.
CP
X
-0.80
-0.50
-0.20
0.10
0.40
0.70
1.00
-1.00
-0.60
-0.20
0.20
0.60
1.00

Figure 3.7

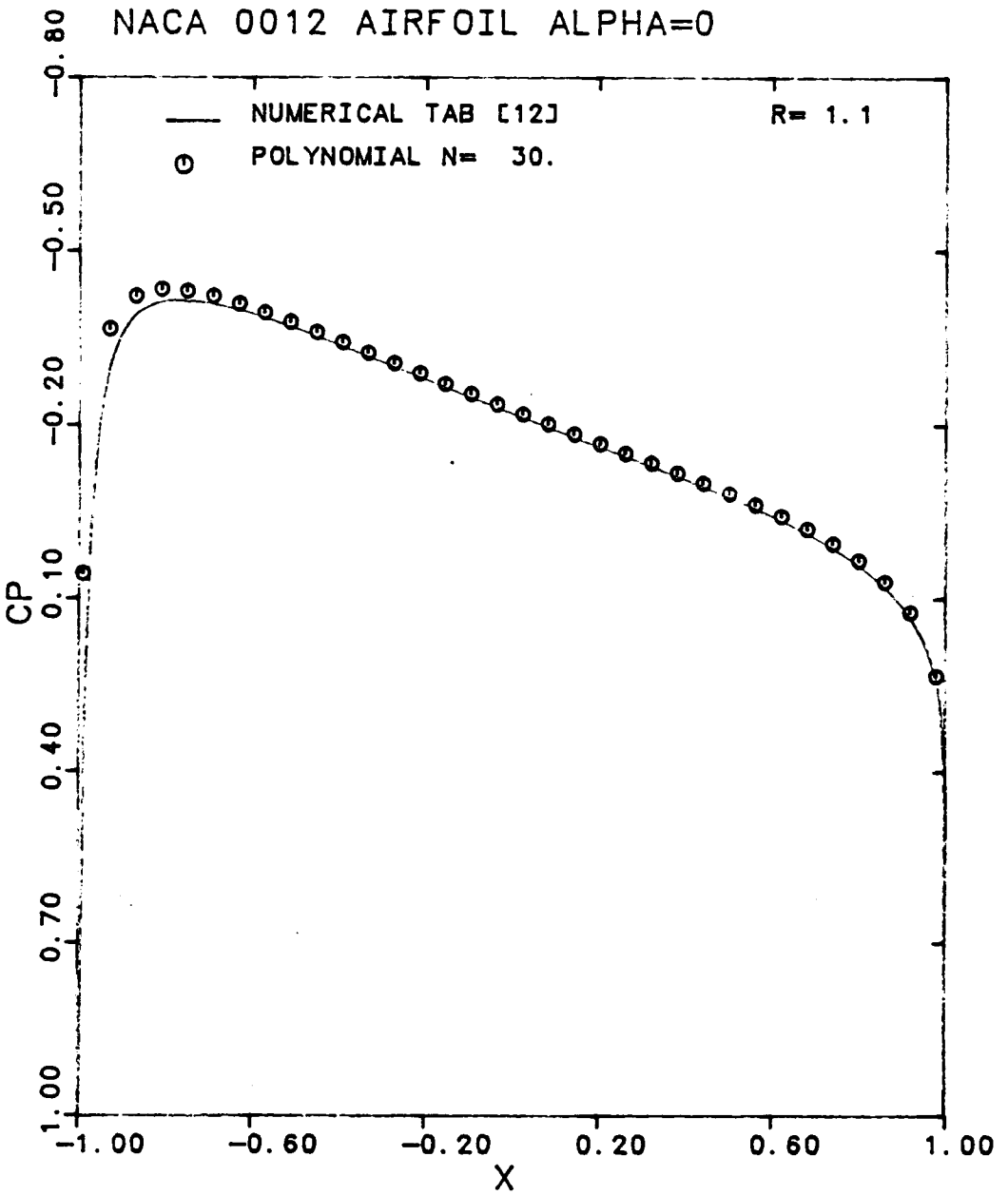

NACA 0012 AIRFOIL ALPHA=0
NUMERICAL TAB [12]
R= 1.1
POLYNOMIAL N= 30.
CP
X
-0.80
-0.50
-0.20
0.10
0.40
0.70
1.00
-1.00
-0.60
-0.20
0.20
0.60
1.00

Figure 3.8

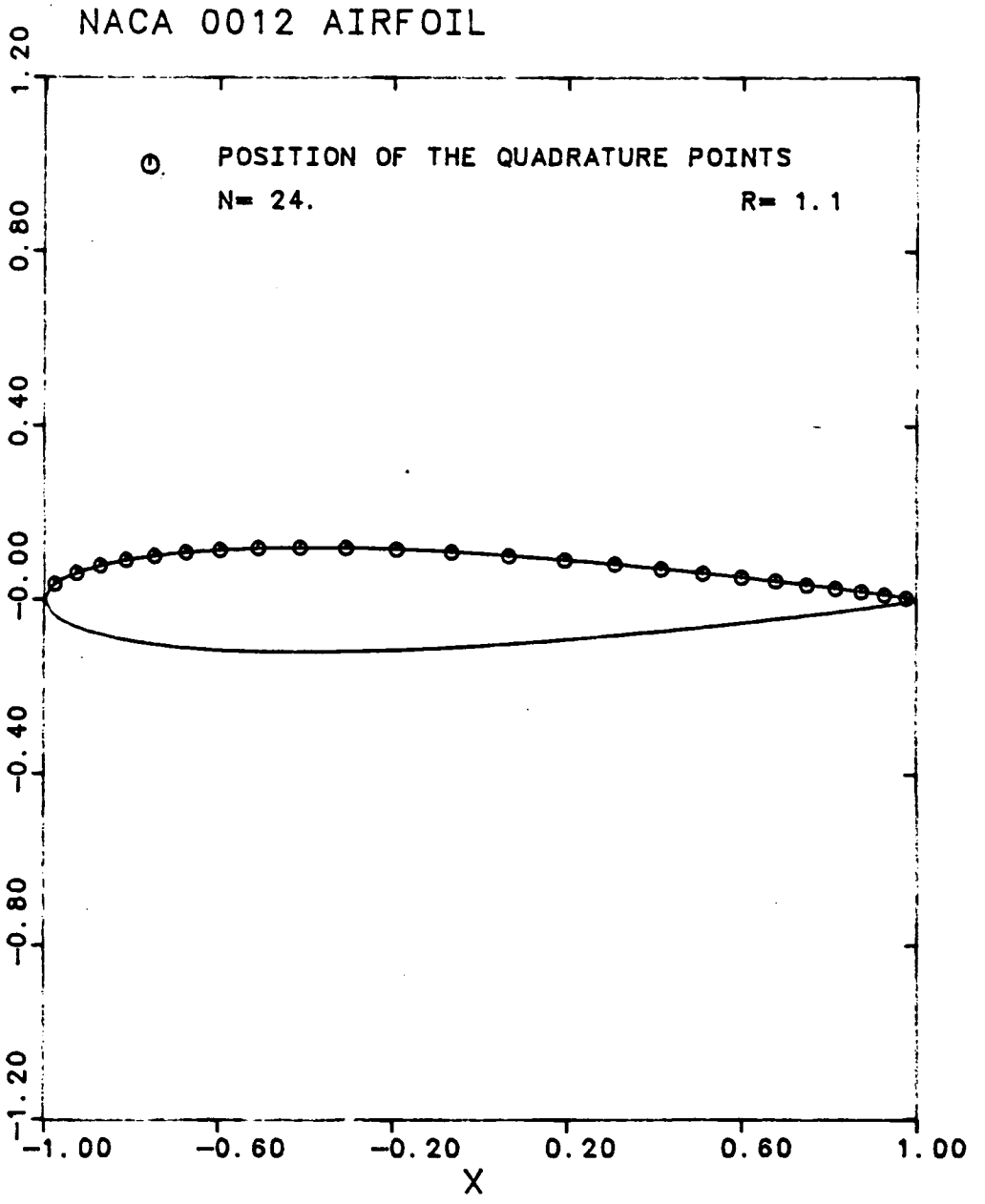

NACA 0012 AIRFOIL
POSITION OF THE QUADRATURE POINTS
N= 24.
R= 1.1
1.20
0.80
0.40
-0.00
-0.40
-0.80
-1.20
-1.00
-0.60
-0.20
0.20
0.60
1.00
X

Figure 3.9

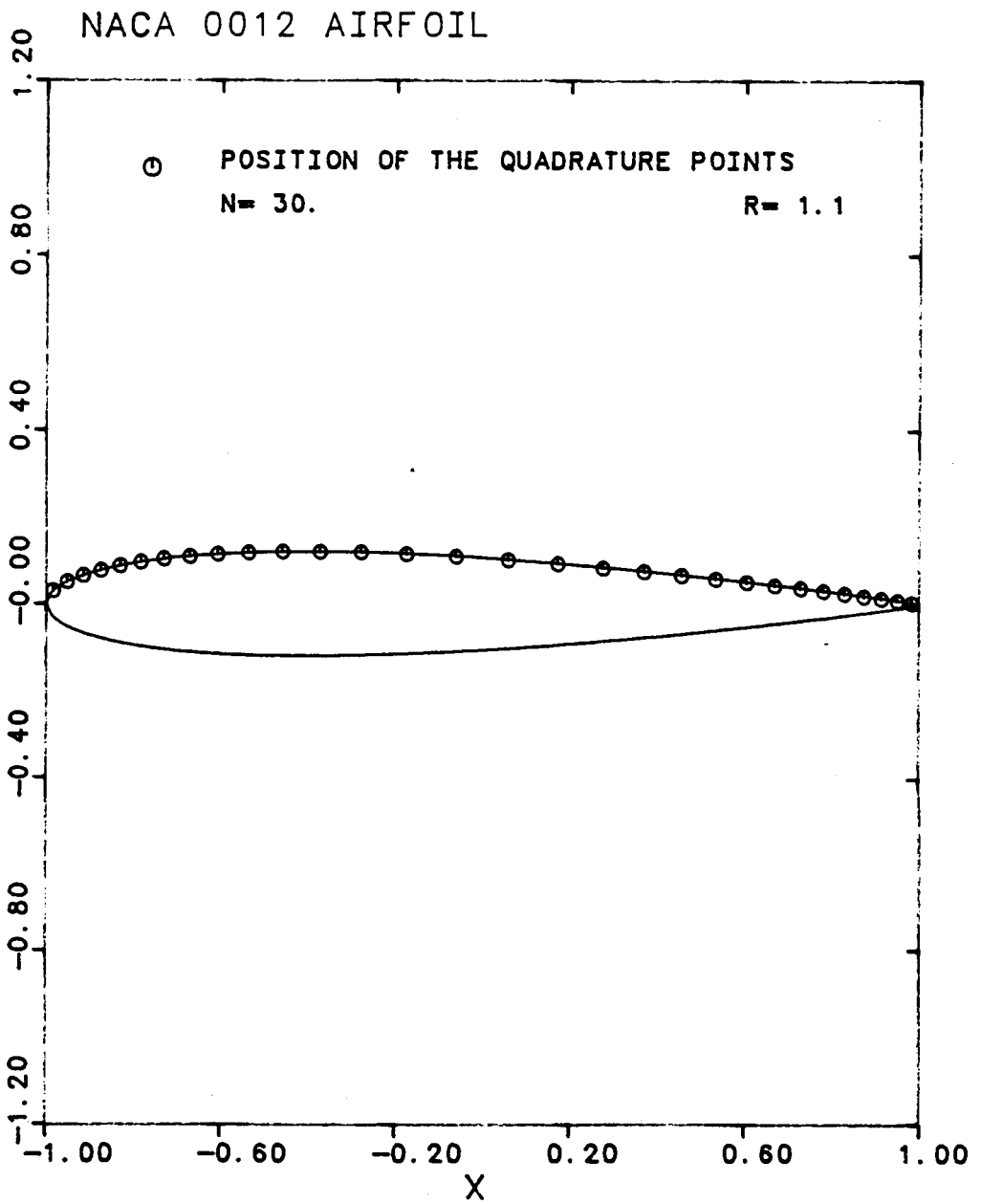

NACA 0012 AIRFOIL
POSITION OF THE QUADRATURE POINTS
N= 30.
R= 1.1
1.20
0.80
0.40
-0.00
-0.40
-0.80
-1.20
-1.00
-0.60
-0.20
0.20
0.60
1.00
X

Figure 3.10

3.5 NACA 0012 at 4 and 10 Degrees Angle of Attack

Figures 3.11, 3.12 and 3.13 show pressure distributions on the airfoil at 4 degrees angle of attack and figure 3.14 shows calculated results for 10 degrees. Two curves are shown for each figure. One corresponds to the pressure distribution on the lower plane and the other is the pressure distribution on the upper plane. It should be remembered that the distributions for both the lower plane and the upper plane are independantly calculated and the solutions are matched along the x-axis upstream and downstream of the airfoil.

In figure 3.11, a equally spaced grid has been used while in figures 3.12 to 3.19 a geometrically increasing grid has been used. In figure 3.11, the calculated distribution and the distribution obtained from the NASA code [9] are virtually identical. Again agreement with the NASA code is excellent in figures 3.12 and 3.13. Note that for a slight change of angle of attack from 0 to 4 degrees, the maximum peak of the pressure distribution experiences a change from approximately -0.5 to -1.5. For 10 degrees angle of attack shown in figures 3.14 and 3.15, the upper plane calculation gives reasonably accurate results, while the lower plane calculation gives almost identical results compared to the data [13]. Positions of the collocation points are given by the variable grid scheme described in section 3.2 . The calculated **Cp** is slightly less

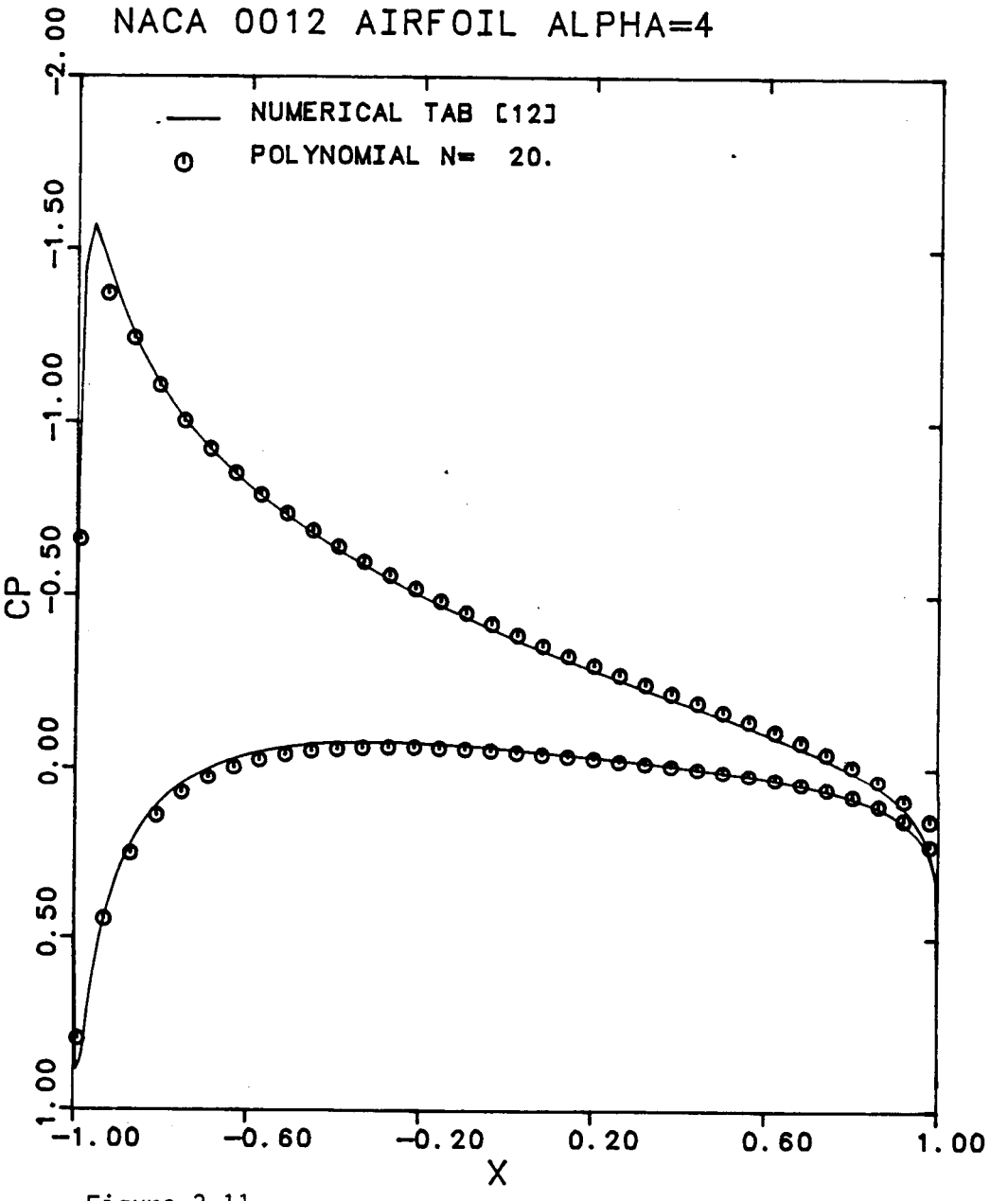

NACA 0012 AIRFOIL ALPHA=4
NUMERICAL TAB [12]
POLYNOMIAL N= 20.
CP
X
-2.00
-1.50
-1.00
-0.50
0.00
0.50
1.00
-1.00
-0.60
-0.20
0.20
0.60
1.00

Figure 3.11

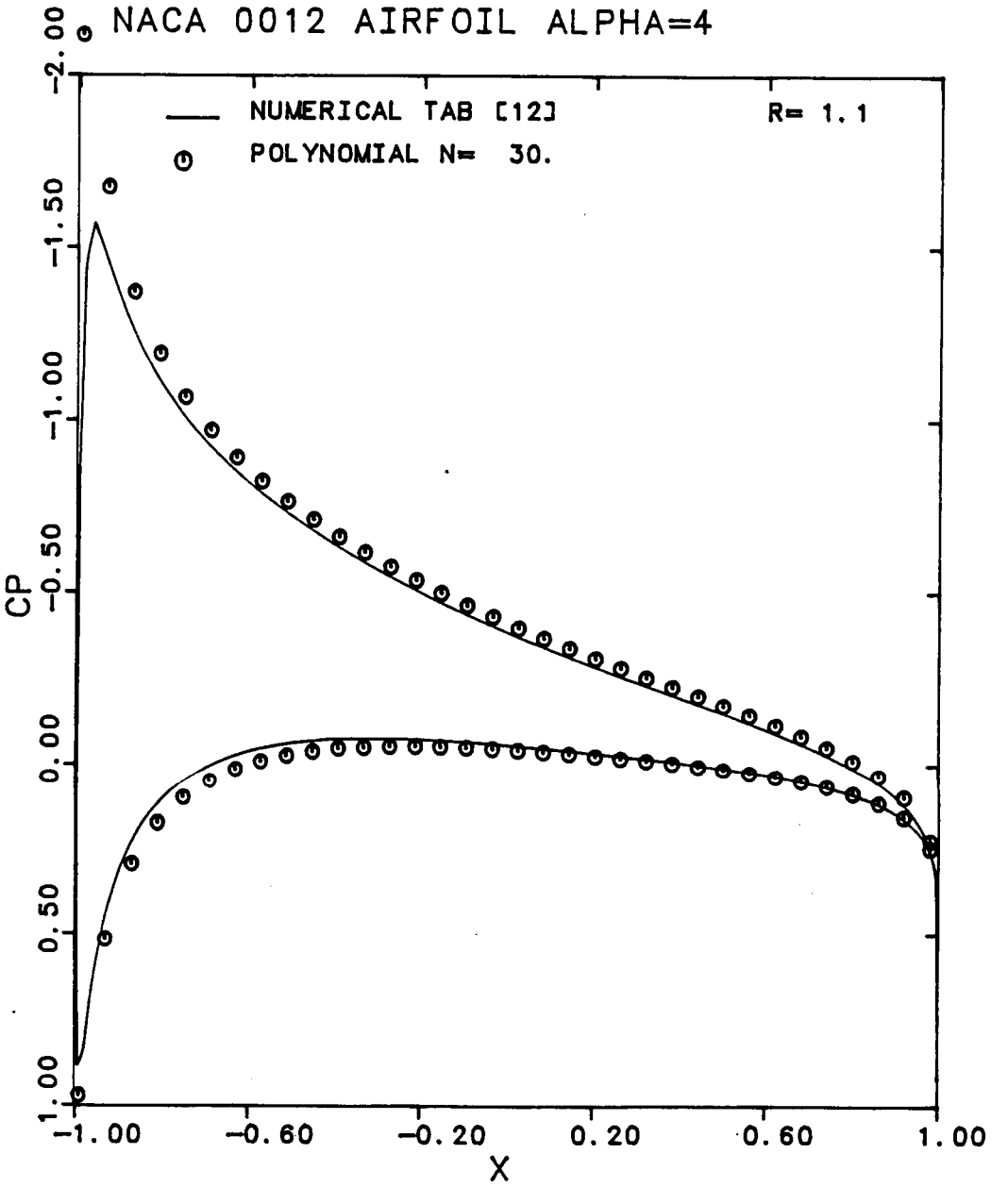

NACA 0012 AIRFOIL ALPHA=4
NUMERICAL TAB [12]
POLYNOMIAL N= 30.
R= 1.1
CP
X
-2.00
-1.50
-1.00
-0.50
0.00
0.50
1.00
-1.00
-0.60
-0.20
0.20
0.60
1.00

Figure 3.12

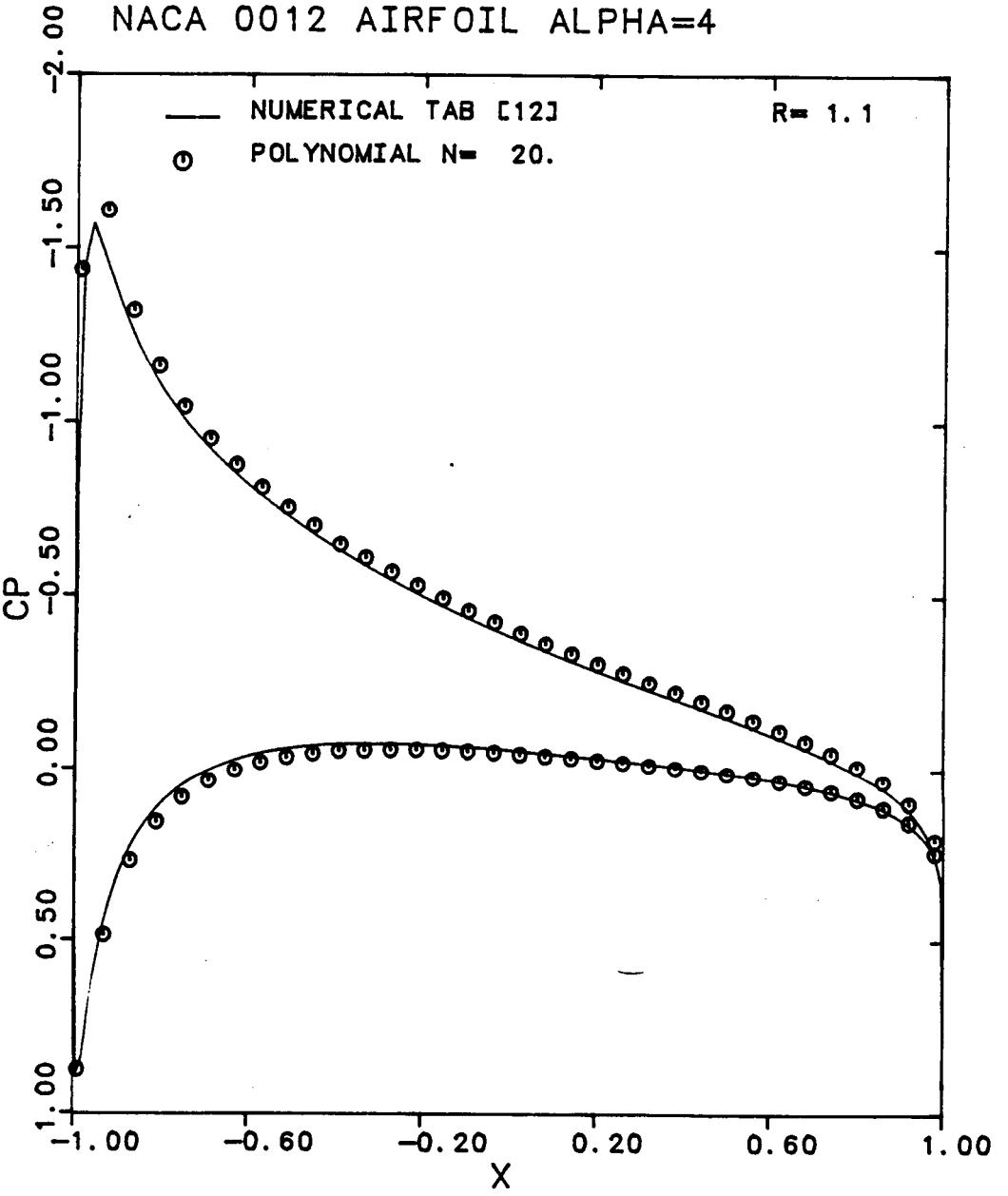

NACA 0012 AIRFOIL ALPHA=4
NUMERICAL TAB [12]
R= 1.1
POLYNOMIAL N= 20.
CP
X
-2.00
-1.50
-1.00
-0.50
0.00
0.50
1.00
-1.00
-0.60
-0.20
0.20
0.60
1.00

Figure 3.13

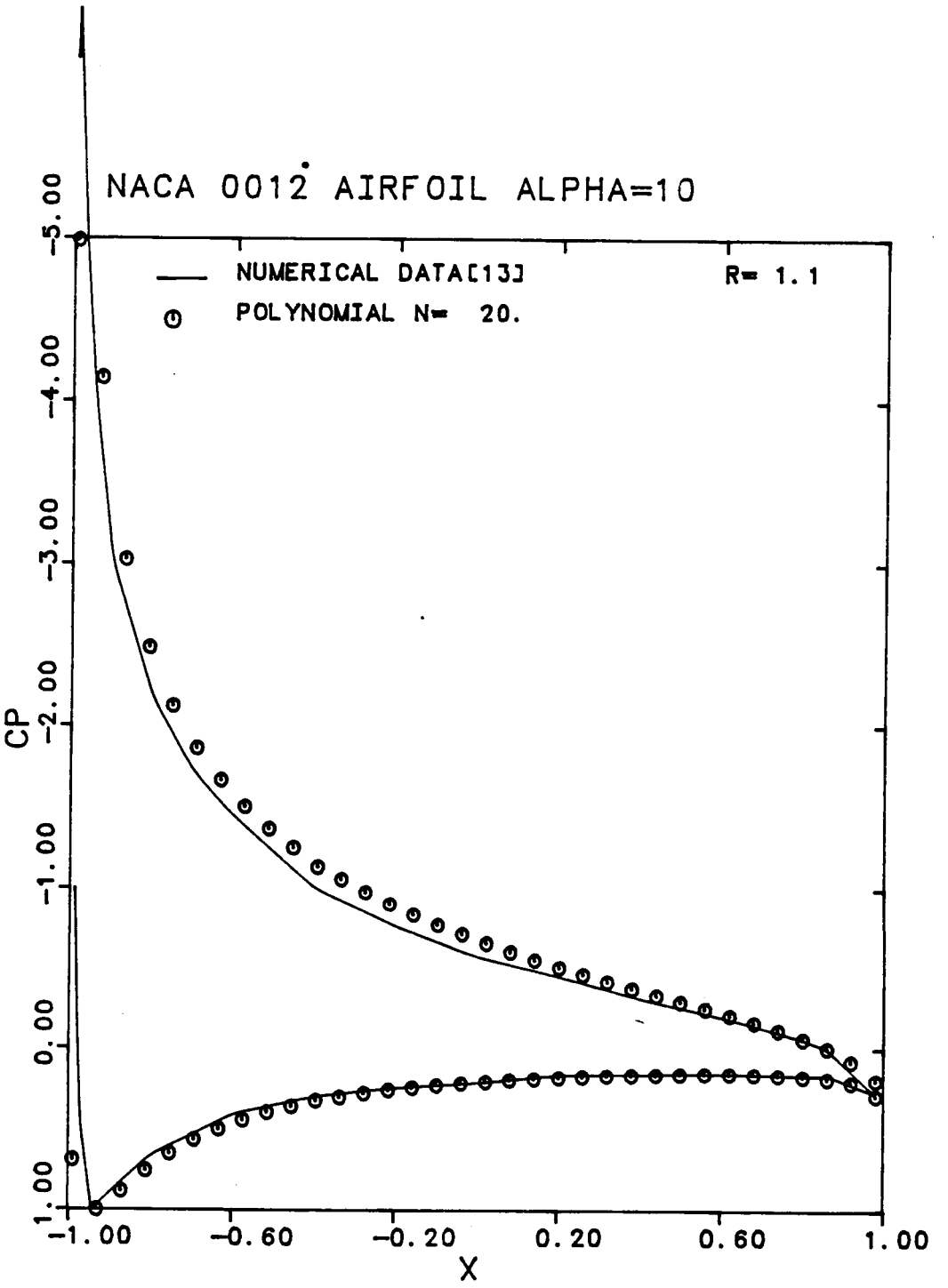

NACA 0012 AIRFOIL ALPHA=10
NUMERICAL DATA[13]
R= 1.1
POLYNOMIAL N= 20.
CP
X
-5.00
-4.00
-3.00
-2.00
-1.00
0.00
1.00
-1.00
-0.60
-0.20
0.20
0.60
1.00

Figure 3.14

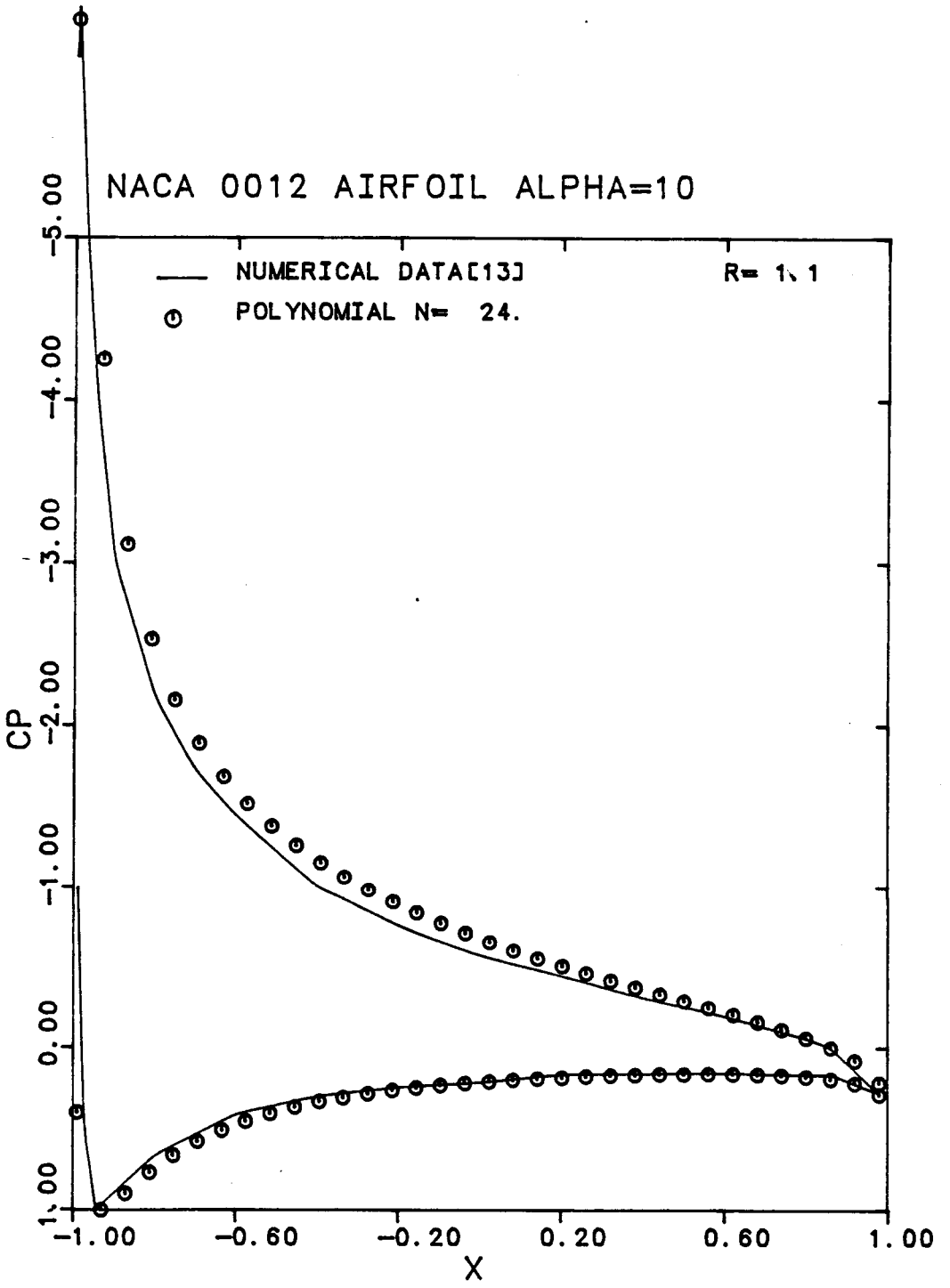

NACA 0012 AIRFOIL ALPHA=10
NUMERICAL DATA[13]
POLYNOMIAL N= 24.
R= 1.1
CP
X
-5.00
-4.00
-3.00
-2.00
-1.00
0.00
1.00
-1.00
-0.60
-0.20
0.20
0.60
1.00

Figure 3.15

than the reference data in most of the upper central plane region. It should be noted that the trailing and leading edge regions are in close agreement with the data [9].

3.6 Joukowski Airfoil

Figure 3.16 shows the computed pressure distribution for the case of the symmetrical Joukowski airfoil in a steady flow at a zero degree angle of attack. A comparison is made with the calculated pressure distribution obtained by using a transformed plane and the Joukowski transformation. Details about the calculation procedure can be found in reference [14].

The polynomial surface pressure distribution deviates quite seriously over the region of the negative high pressure peak, toward the leading edge. The correlation is quite reasonable on the surface of the right half airfoil plane toward the trailing edge and the general trend of the C_p polynomial curve is in good agreement with the calculated curve. The main problem occurs in the region of the negative high pressure peak where the pressure distribution is overpredicted. Extensive calculations were made in an effort to improve the agreement. Different polynomial approximation schemes were used and the analytical solutions were carefully checked. In all cases, the present method consistently overpredicted the negative pressure peak. It has been concluded that this error is probably

due to the error introduced in the conformal mapping which produces a slightly displaced leading edge. As noted in section 2.4.2, this error is approximatly 1.5% for the 12% thick Joukowski airfoil. It should be noted, that similar disagreements have been observed by other investigators [12].

Figures 3.17, 3.18 and 3.19 show pressure distribution calculations made for different grid point distributions. Positions of the points are shown in figures 3.20 and 3.21. An important aspect is that the computation procedure gives consistent results using different grid point distributions.

The essential difference between a cusped trailing edge and a trailing edge of finite angle is evident from a comparison of figure 3.7 (NACA 0012 airfoil) and figure 3.18 (Joukowski airfoil). The behavior of the flow at the trailing edge is accuratly calculated by the polynomial method as shown in figure 3.18 and the correlation for both the polynomial and theoretical curves agree well in the trailing edge region.

The calculations for the flow about the symmetrical Joukowski airfoil of thickness 12% were repeatead at 4 and 10 degrees of angle of attack. The calculated pressure distributions are compared with the analytic solution in figures 3.22 and 3.23. For the lower curve of figure 3.22, both pressure distributions are virtually identical. The largest disagreement occurs near the negative high pressure peak.

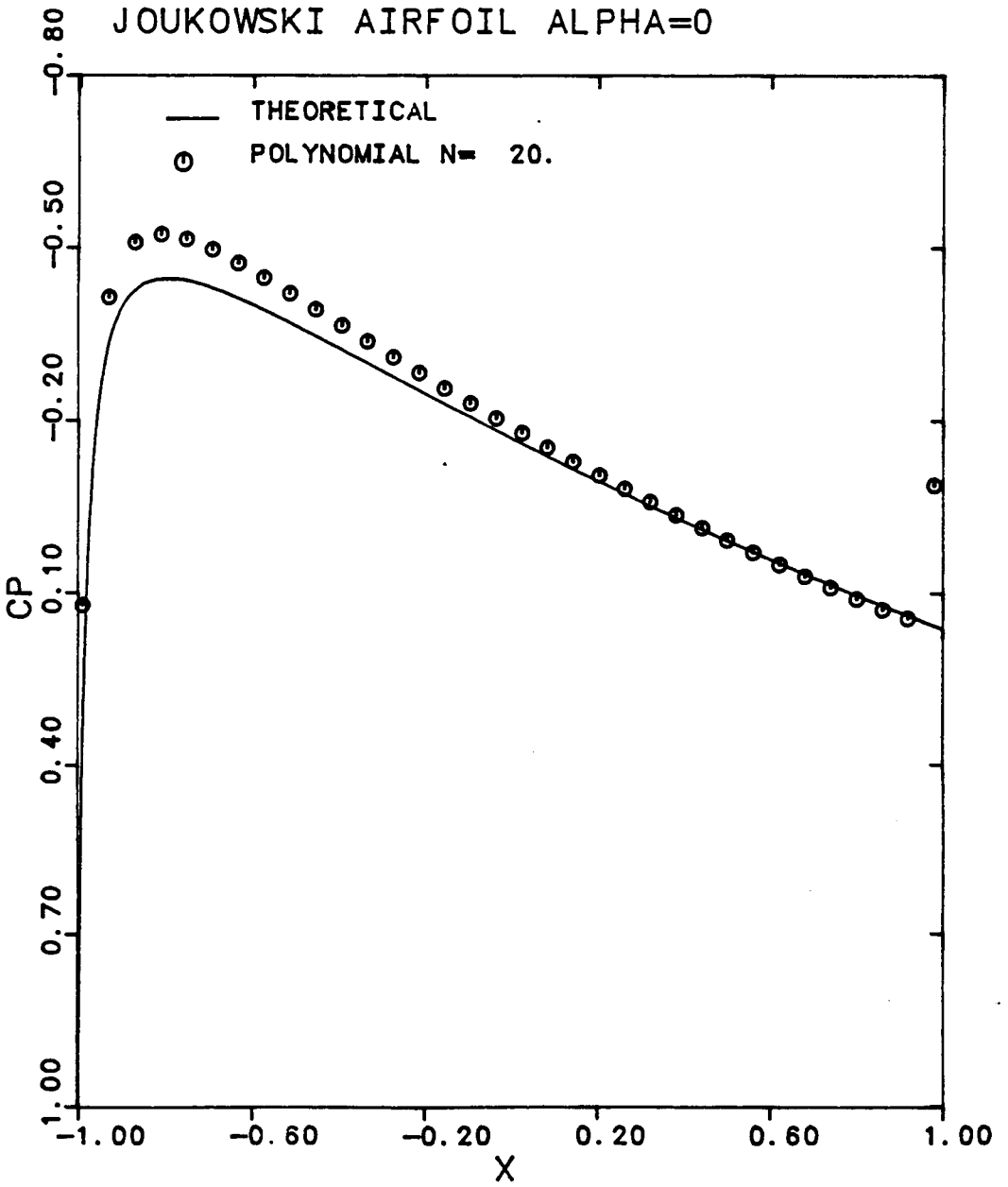

JOUKOWSKI AIRFOIL ALPHA=0
THEORETICAL
POLYNOMIAL N= 20.
CP
X
-0.80
-0.50
-0.20
0.10
0.40
0.70
1.00
-1.00
-0.60
-0.20
0.20
0.60
1.00

Figure 3.16

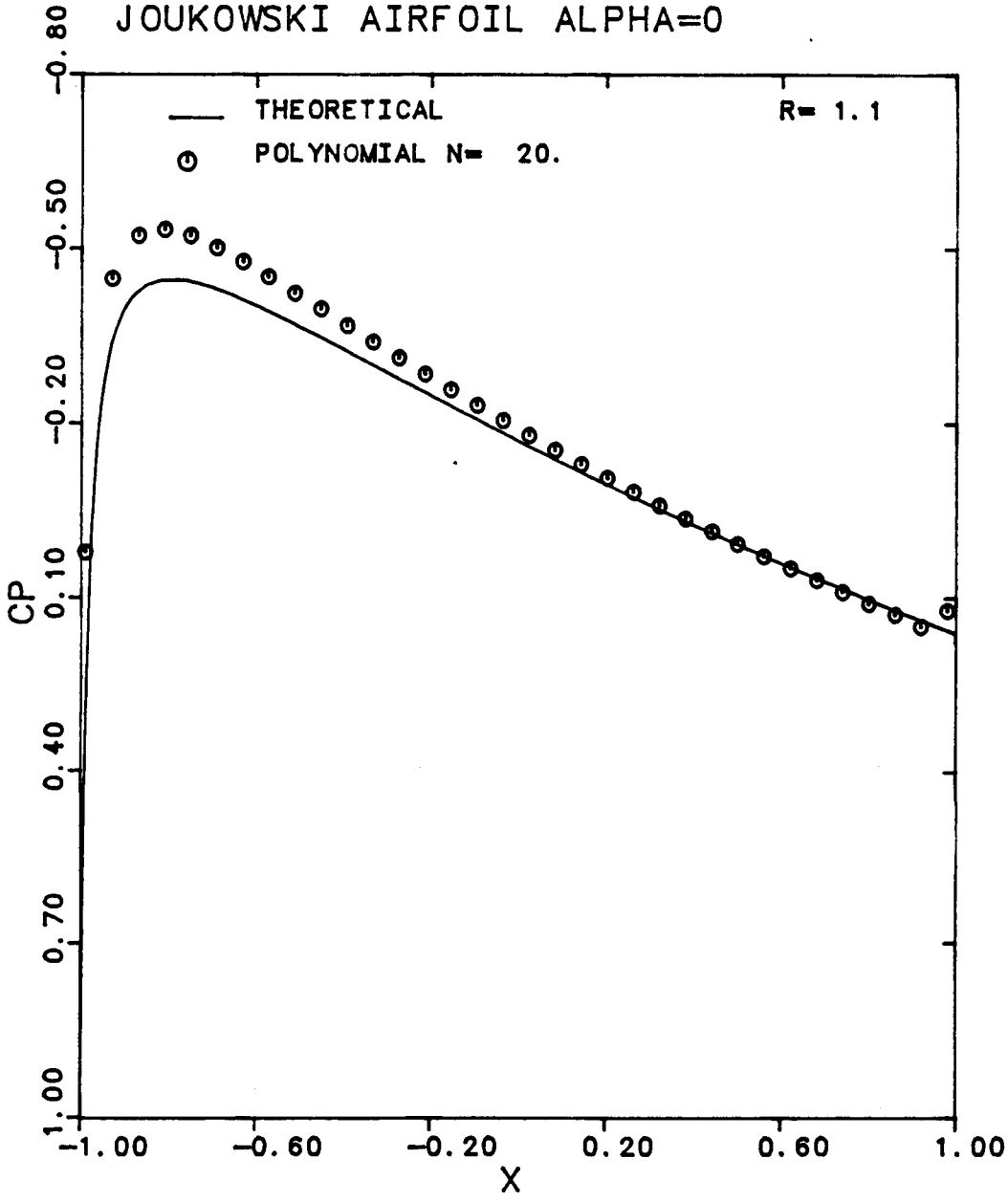

JOUKOWSKI AIRFOIL ALPHA=0
THEORETICAL
POLYNOMIAL N= 20.
R= 1.1
CP
X
-0.80
-0.50
-0.20
0.10
0.40
0.70
1.00
-1.00
-0.60
-0.20
0.20
0.60
1.00

Figure 3.17

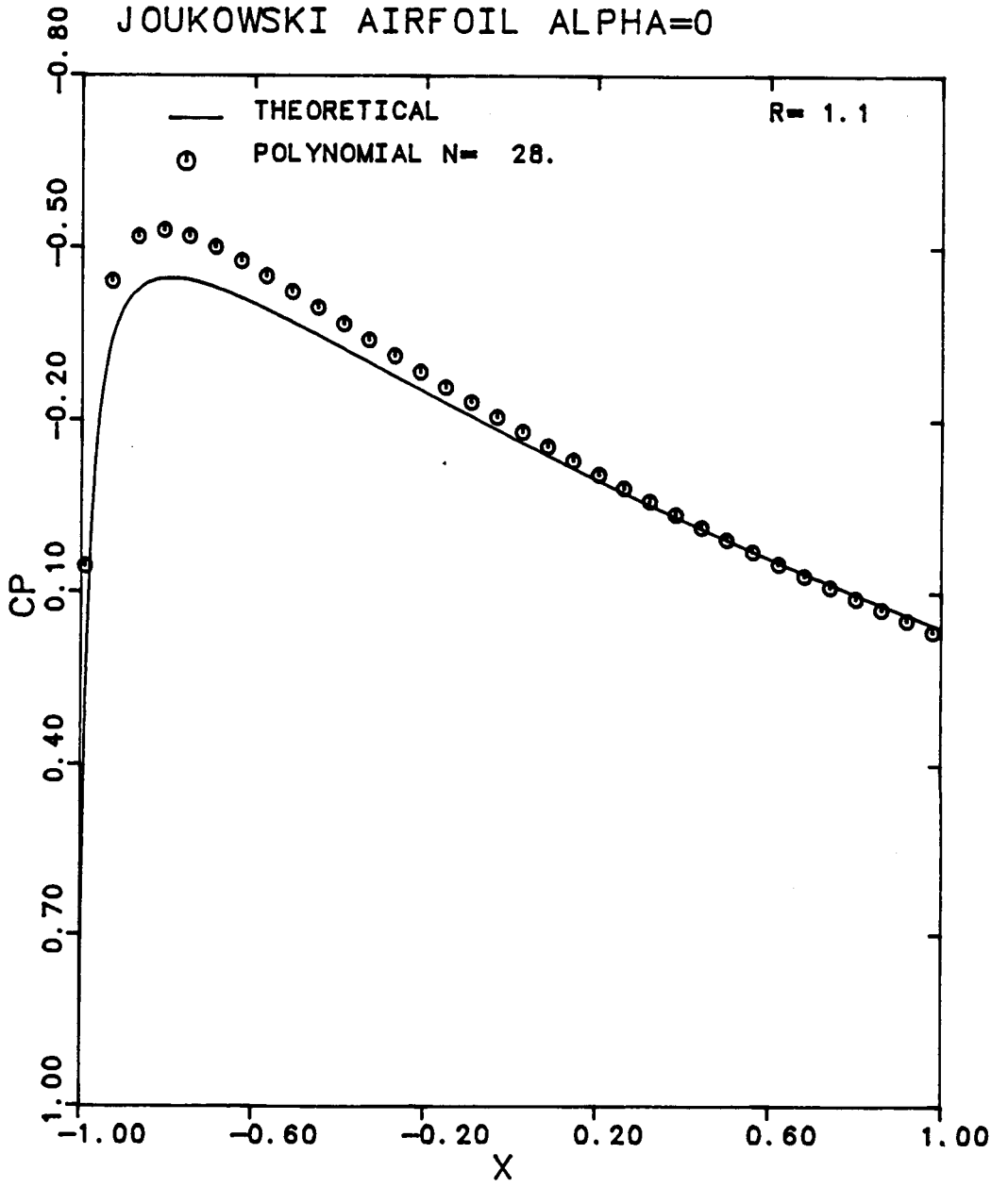

JOUKOWSKI AIRFOIL ALPHA=0
THEORETICAL
POLYNOMIAL N= 28.
R= 1.1
CP
X
0.80
0.50
0.20
0.10
0.40
0.70
1.00
-1.00
-0.60
-0.20
0.20
0.60
1.00

Figure 3.18

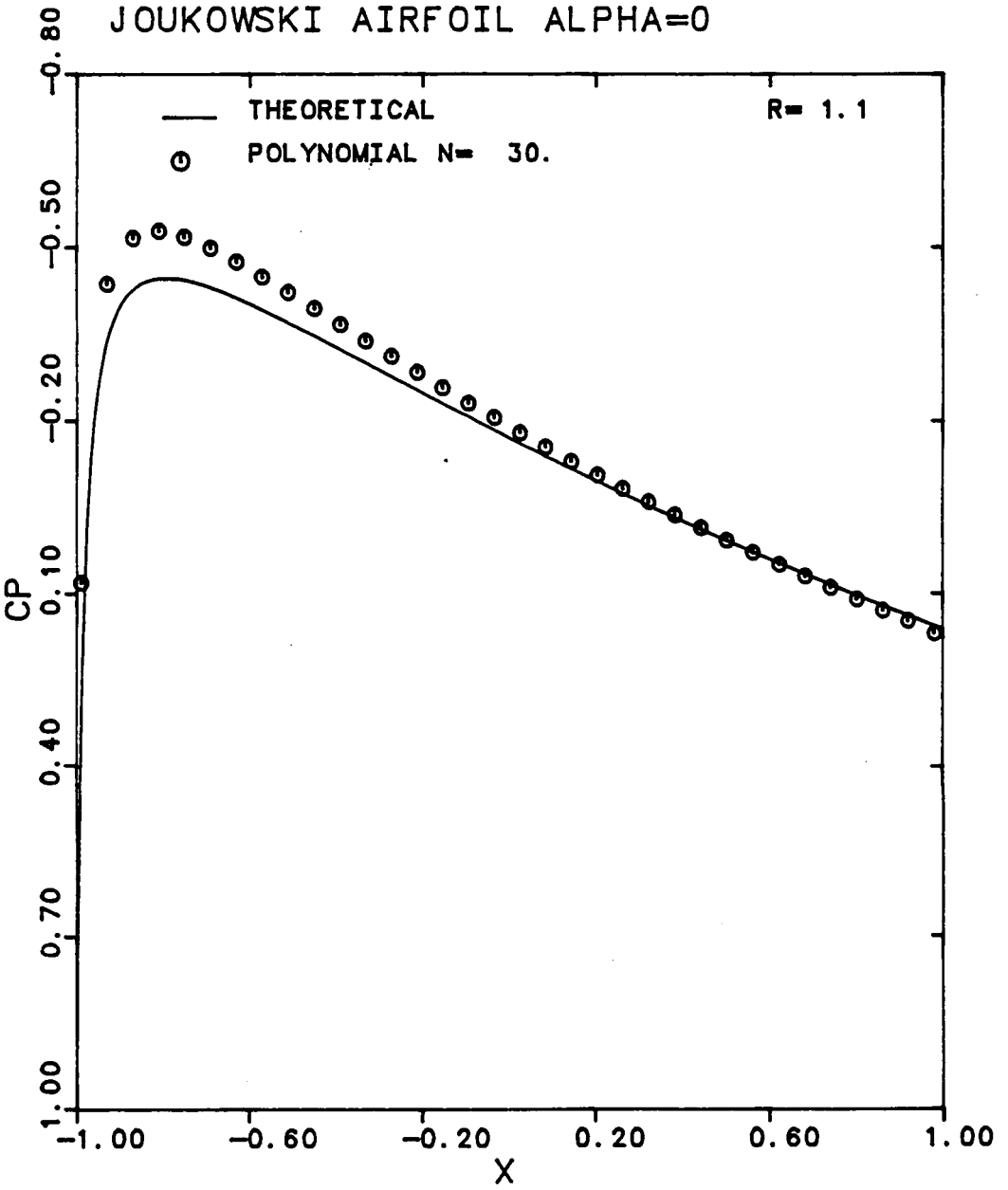

JOUKOWSKI AIRFOIL ALPHA=0
THEORETICAL
POLYNOMIAL N= 30.
R= 1.1
CP
X
-0.80
-0.50
-0.20
0.10
0.40
0.70
1.00
-1.00
-0.60
-0.20
0.20
0.60
1.00

Figure 3.19

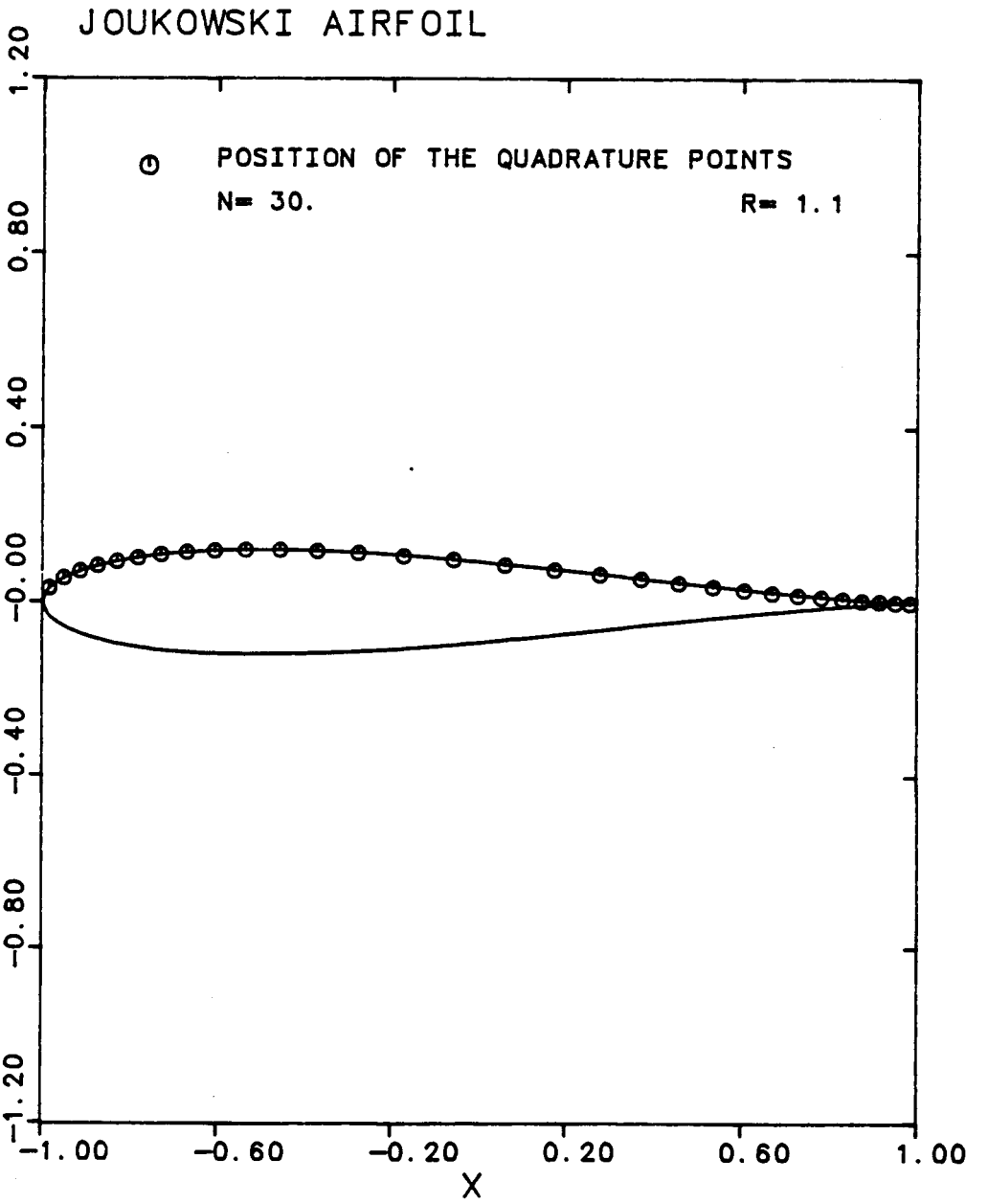

JOUKOWSKI AIRFOIL
POSITION OF THE QUADRATURE POINTS
N= 30.
R= 1.1
1.20
0.80
0.40
-0.00
-0.40
-0.80
-1.20
-1.00
-0.60
-0.20
0.20
0.60
1.00
X

Figure 3.20

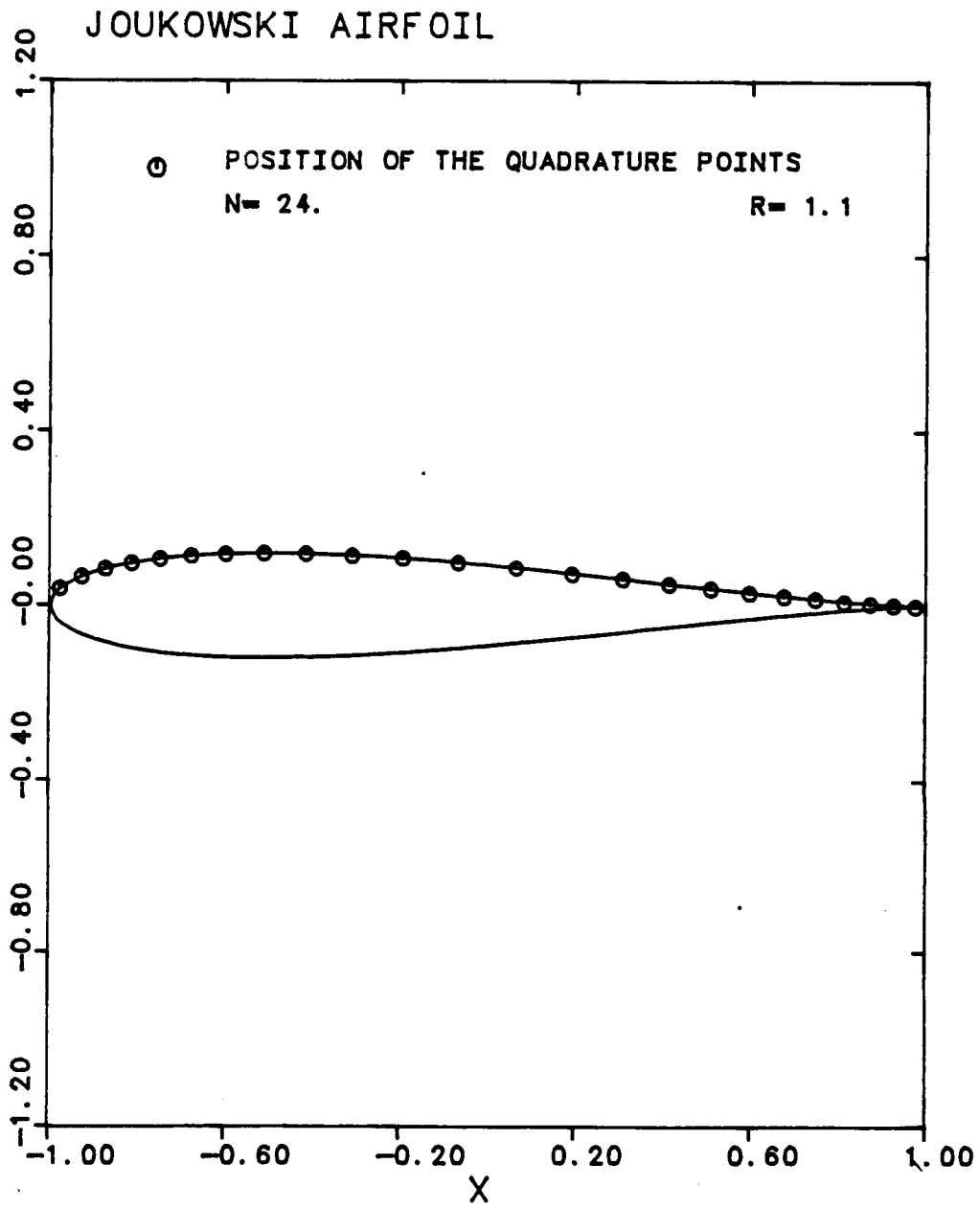

JOUKOWSKI AIRFOIL
POSITION OF THE QUADRATURE POINTS
N= 24.
R= 1.1
1.20
0.80
0.40
-0.00
-0.40
-0.80
-1.20
-1.00
-0.60
-0.20
0.20
0.60
1.00
X

Figure 3.21

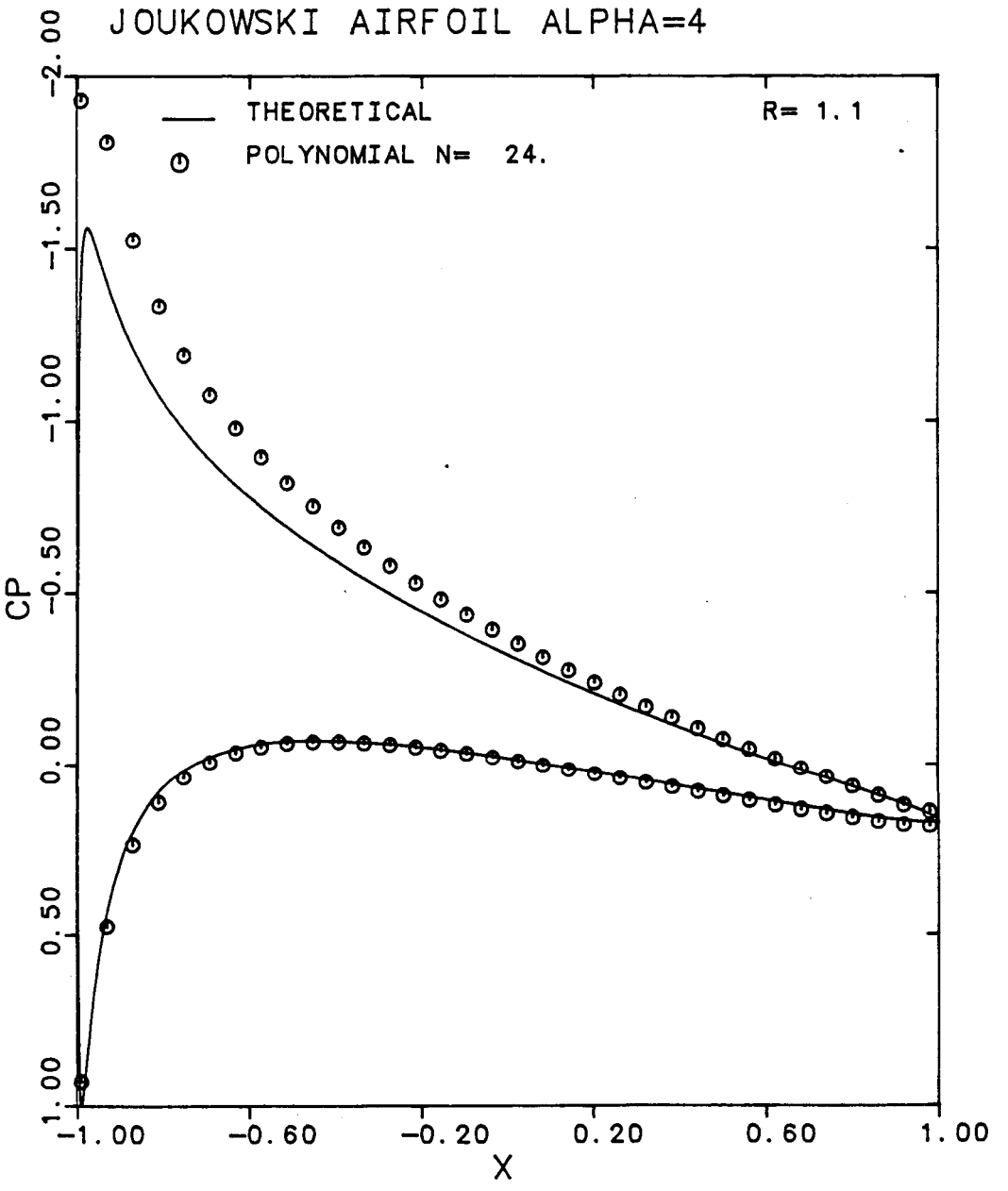

JOUKOWSKI AIRFOIL ALPHA=4
THEORETICAL
POLYNOMIAL N= 24.
R= 1.1
CP
X
-2.00
-1.50
-1.00
-0.50
0.00
0.50
1.00
-1.00
-0.60
-0.20
0.20
0.60
1.00

Figure 3.22

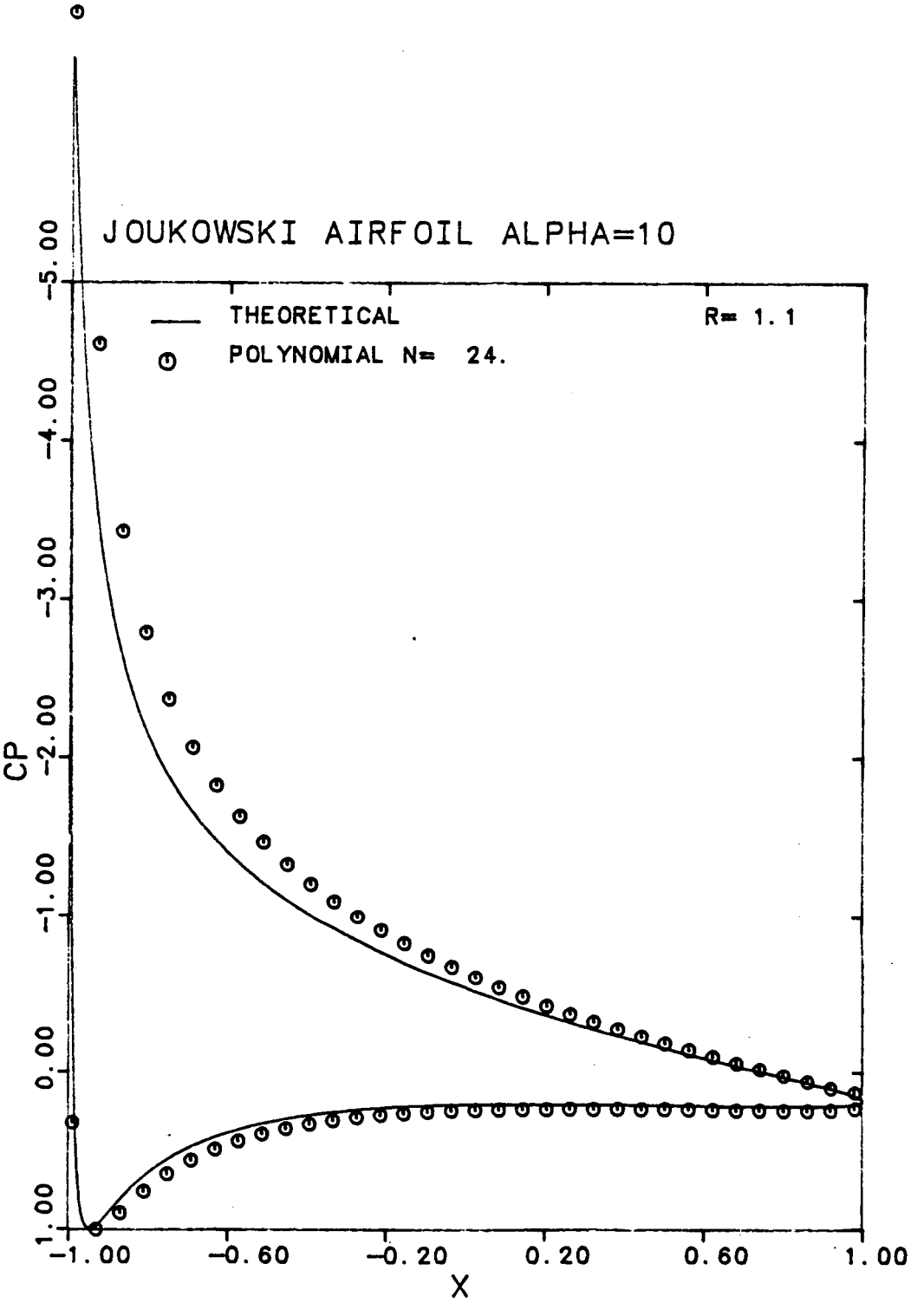

JOUKOWSKI AIRFOIL ALPHA=10
THEORETICAL
POLYNOMIAL N= 24.
R= 1.1
CP
X
-5.00
-4.00
-3.00
-2.00
-1.00
0.00
1.00
-1.00
-0.60
-0.20
0.20
0.60
1.00

Figure 3.23

Nevertheless, it can be seen that the calculated pressure distribution is in close agreement with the theoritical distribution near the trailing edge. Slight changes in the locations of the collocation points do not significantly affect the shape of the polynomial curve.

3.7 Inverse Design Results

The problem of solving for the body shape given the surface velocity distribution uses essentially the same approach as the direct problem. The calculation procedure is described in detail in chapter 2. However, the technique is not based on an iteration technique as most design methods found in the literature.

The surface velocity distributions used were exact solutions for the cylinder, the elliptic airfoil and the Joukowski airfoil and a numerical approximation for the NACA 0012 airfoil. The calculated body shape is compared to the exact body to evaluate the accuracy of the method.

Figures 3.24, 3.25, 3.26 and 3.27 show the Y-component of the surface velocity for the 4 described airfoils. Figures 3.28, 3.29, 3.30 and 3.31 show the calculated and exact shape for the 4 airfoils. Using 24 equally spaced quadrature points, both the calculated and the exact shape are indistinguishable

for the cylinder, the elliptic airfoil and the Joukowski airfoil. In the NACA 0012 airfoil case, the agreements for both curves are quite good. The calculated shape is slightly underestimated in the region of larger thickness. However, we should remember that the surface velocity distribution for the NACA 0012 airfoil is not exact but obtained from a NASA code [12]. The imprecision in the NASA data could cause the small but not negligible errror of the calculated shape. Another consideration is that the error occurs in the high pressure peak region which also presented an error for the direct calculation.

The design calculation presents less error everywhere as compared to the direct calculation and particularly near the leading and trailing edge.

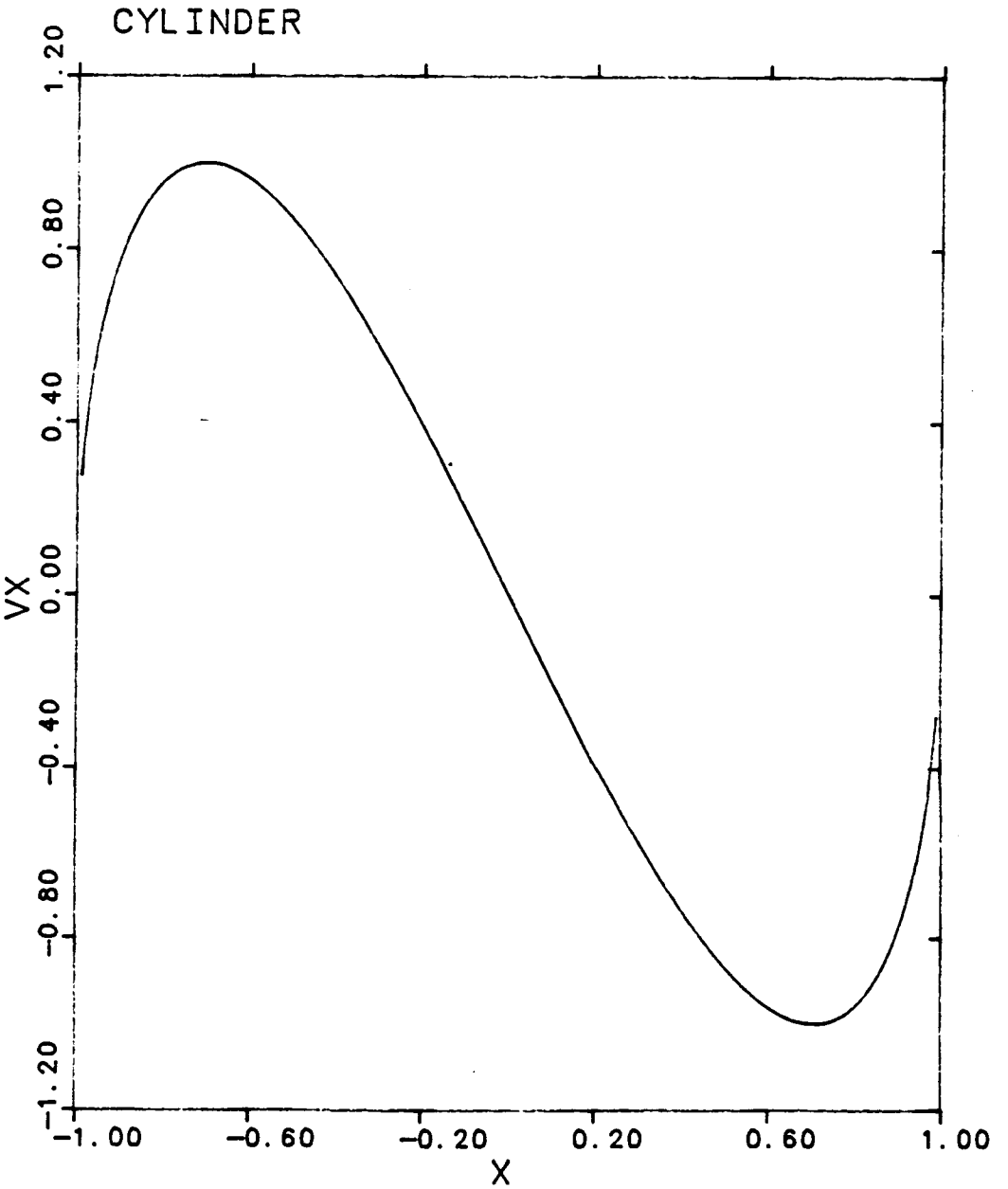

CYLINDER
1.20
0.80
0.40
0.00
-0.40
-0.80
-1.20
VX
-1.00
-0.60
-0.20
0.20
0.60
1.00
X

Figure 3.24

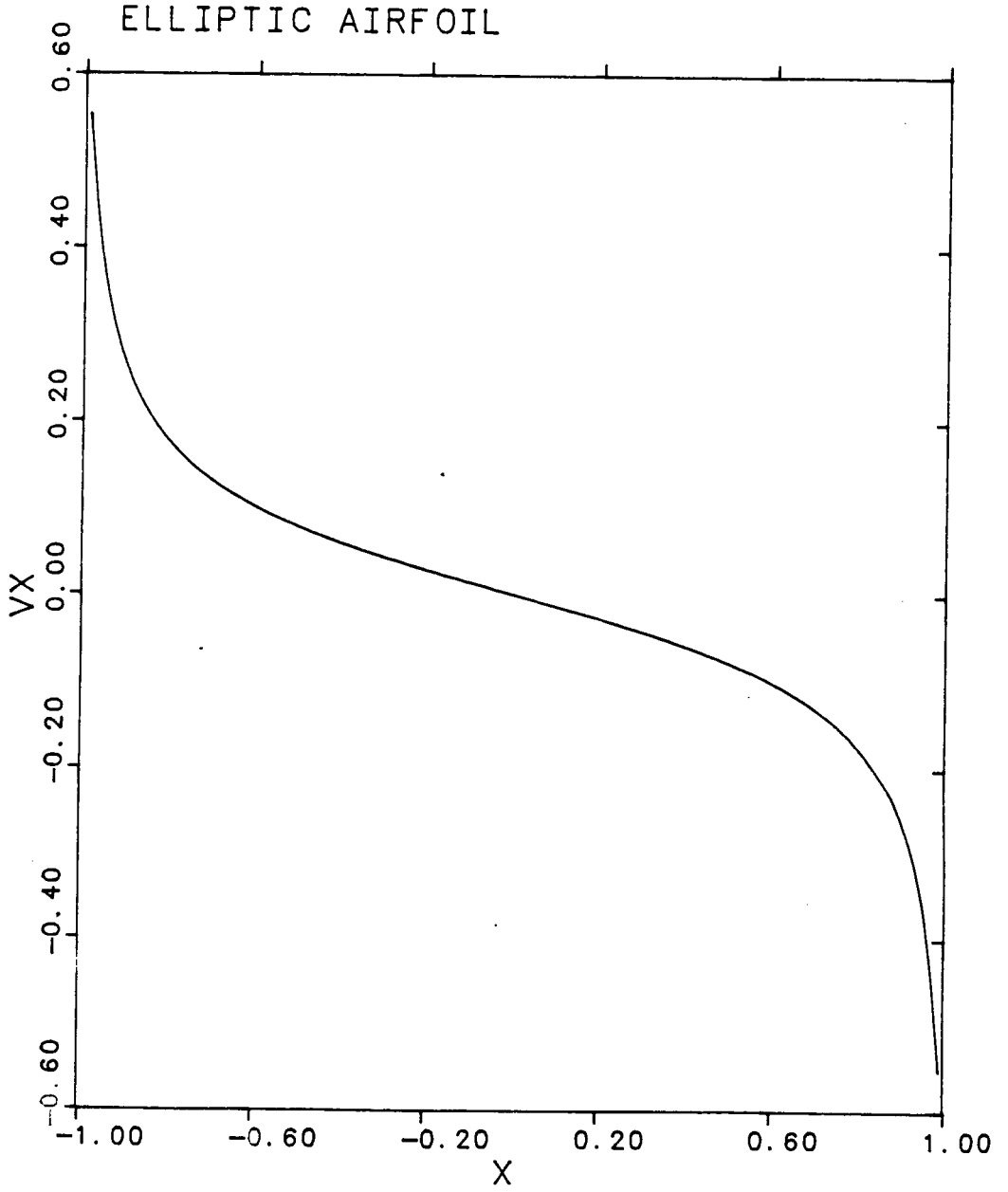

ELLIPTIC AIRFOIL
0.60
0.40
0.20
0.00
-0.20
-0.40
-0.60
VX
-1.00
-0.60
-0.20
0.20
0.60
1.00
X

Figure 3.25

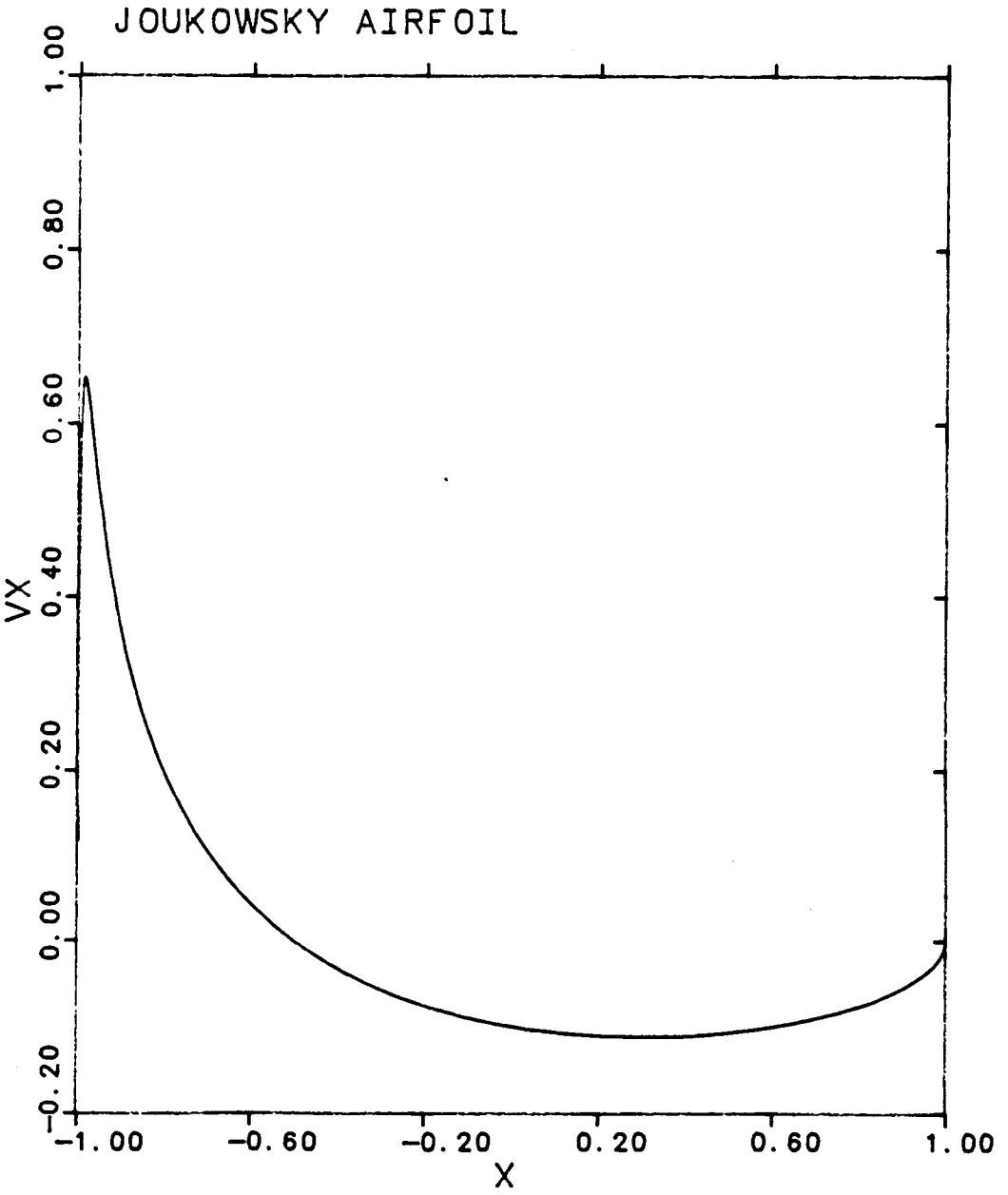

JOUKOWSKY AIRFOIL
1.00
0.80
0.60
0.40
0.20
0.00
-0.20
VX
-1.00
-0.60
-0.20
0.20
0.60
1.00
X

Figure 3.26

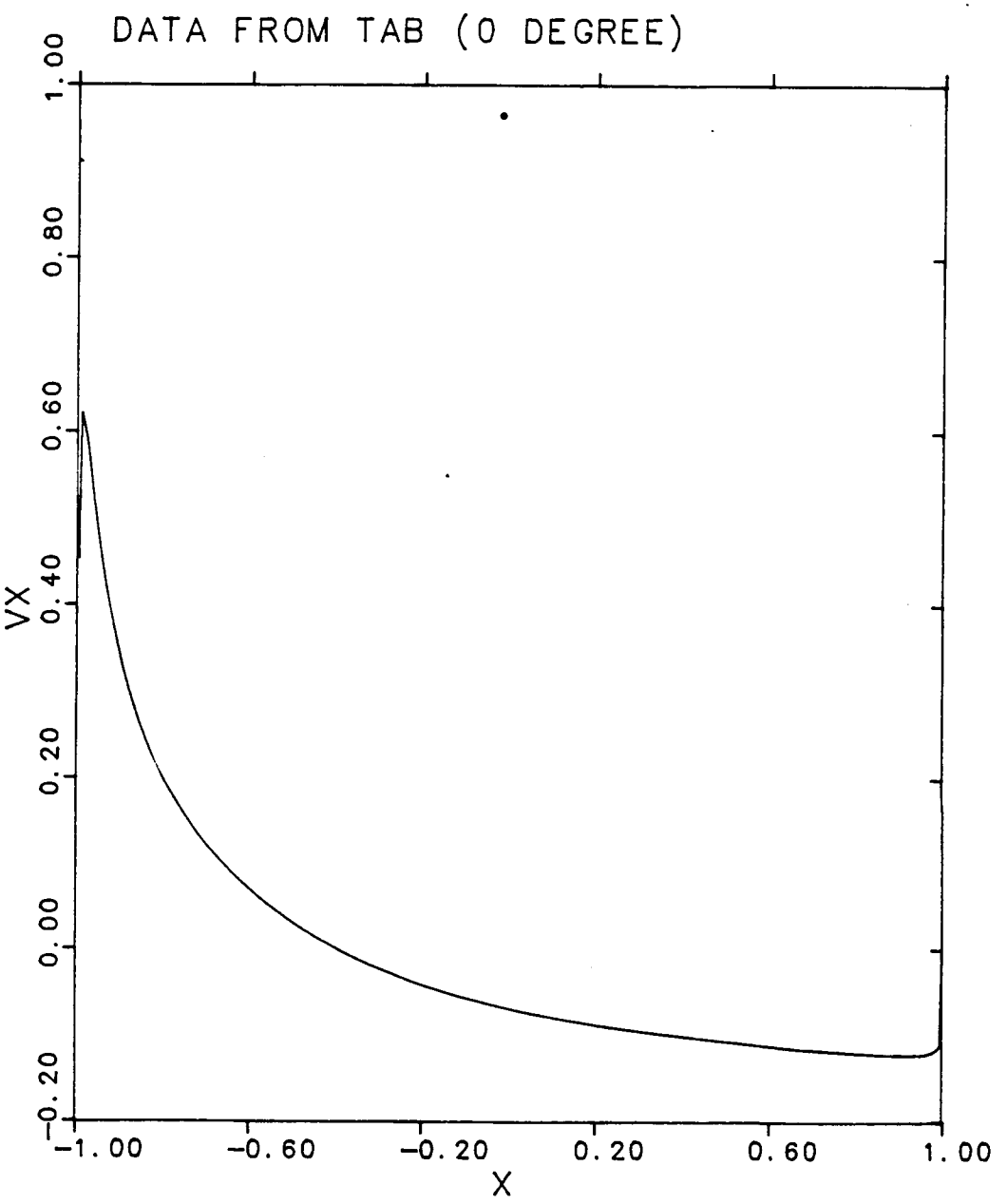

DATA FROM TAB (0 DEGREE)
1.00
0.80
0.60
0.40
0.20
0.00
-0.20
VX
-1.00
-0.60
-0.20
0.20
0.60
1.00
X

Figure 3.27

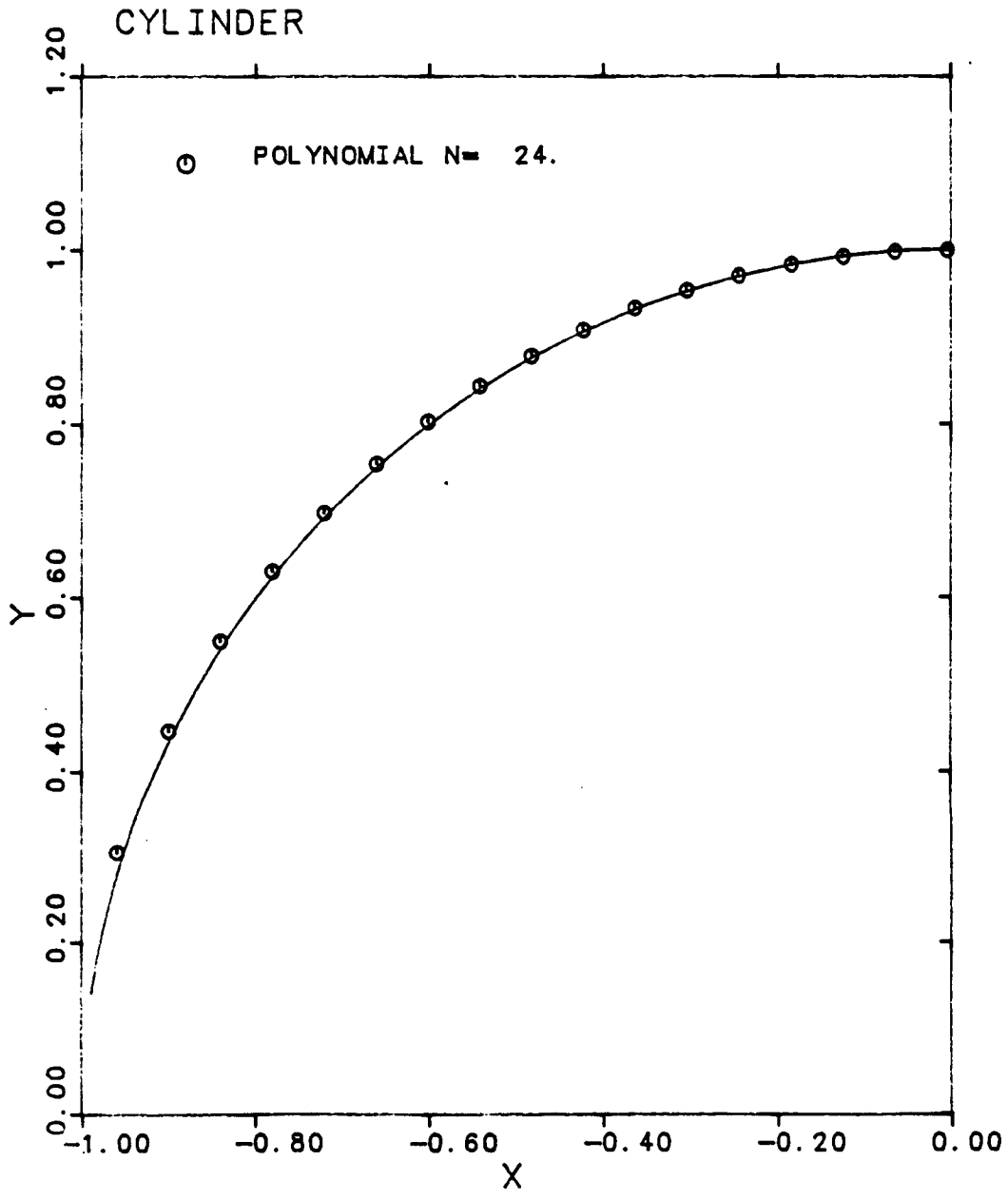

CYLINDER
POLYNOMIAL N= 24.
1.20
1.00
0.80
0.60
0.40
0.20
0.00
Y
-1.00
-0.80
-0.60
-0.40
-0.20
0.00
X

Figure 3.28

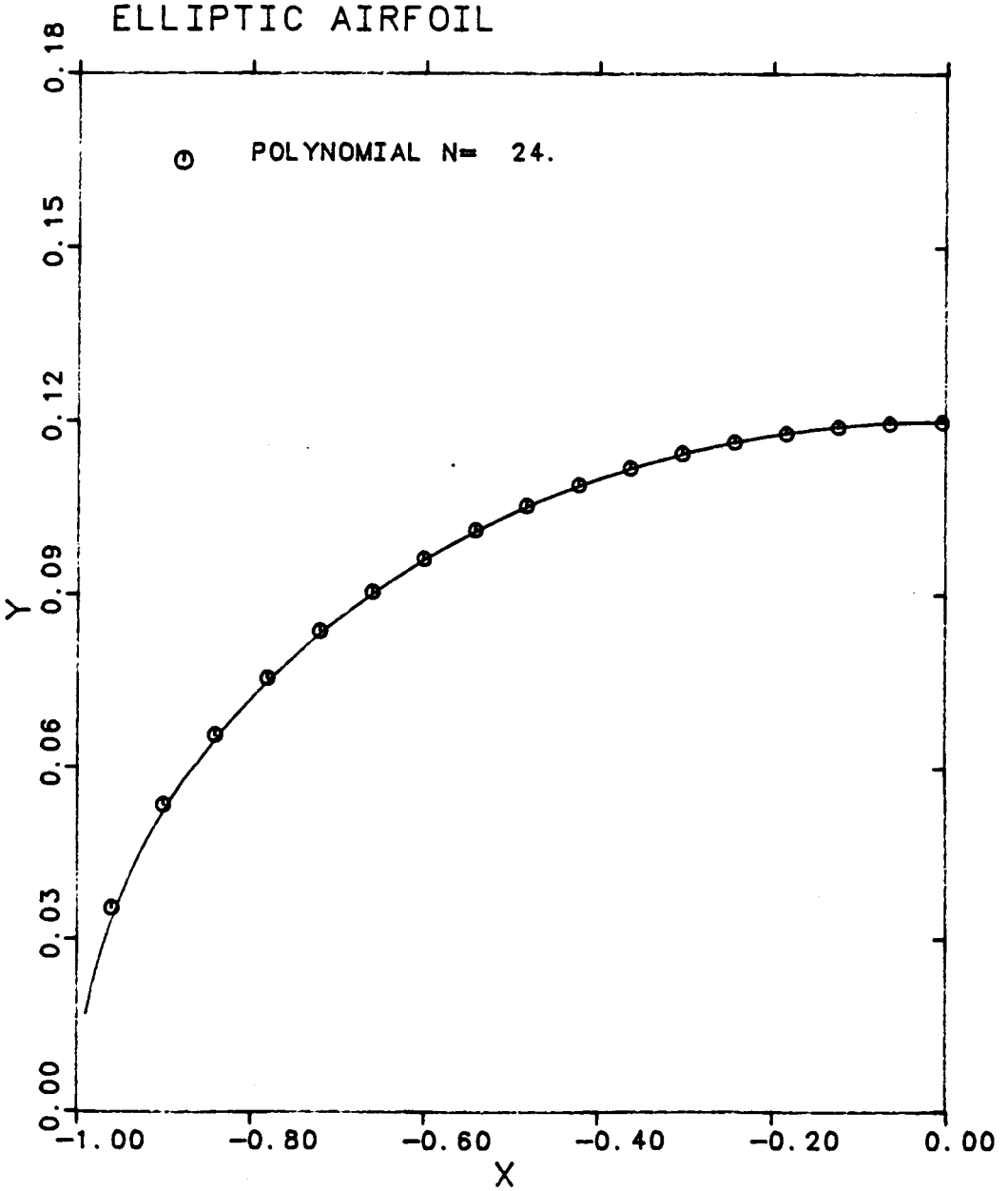

ELLIPTIC AIRFOIL
POLYNOMIAL N= 24.
0.18
0.15
0.12
0.09
0.06
0.03
0.00
Y
-1.00
-0.80
-0.60
-0.40
-0.20
0.00
X

Figure 3.29

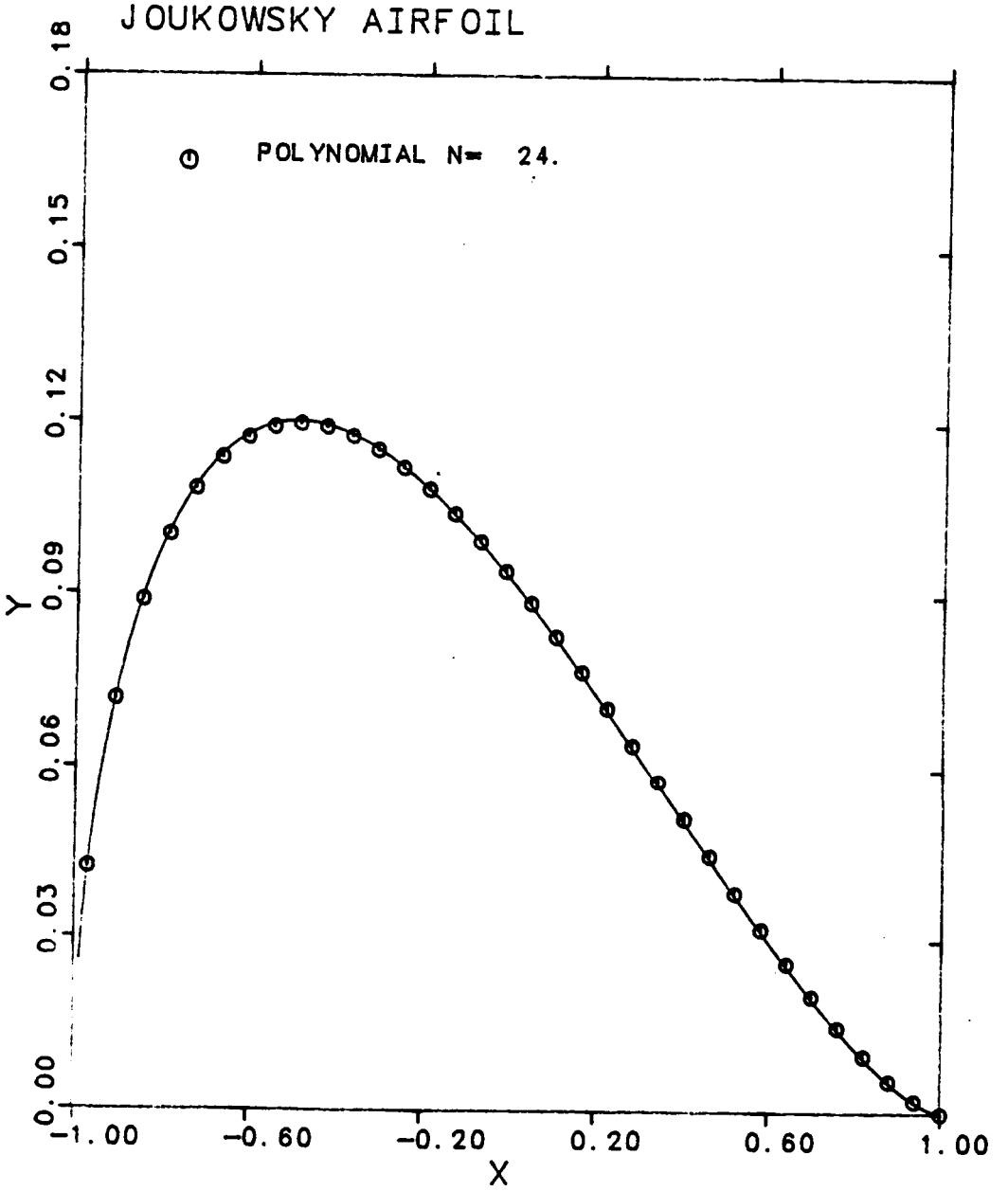

JOUKOWSKY AIRFOIL
POLYNOMIAL N= 24.
Y
X
0.18
0.15
0.12
0.09
0.06
0.03
0.00
-1.00
-0.60
-0.20
0.20
0.60
1.00

Figure 3.30

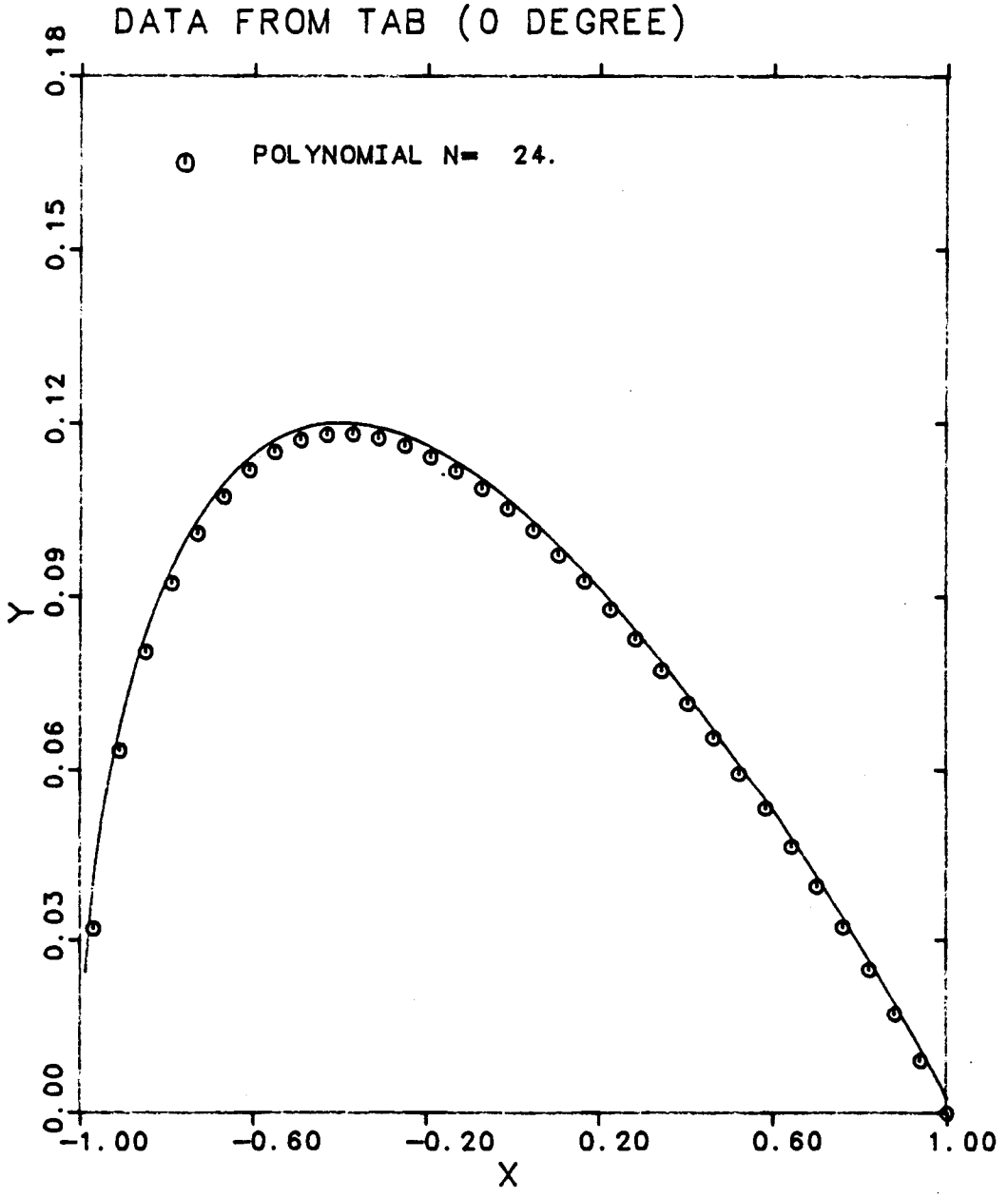

DATA FROM TAB (0 DEGREE)
POLYNOMIAL N= 24.
Y
X
0.18
0.15
0.12
0.09
0.06
0.03
0.00
-1.00
-0.60
-0.20
0.20
0.60
1.00

Figure 3.31

CHAPTER 4
CORRECTION FACTOR FOR THE COMPRESSIBLE CALCULATION

4.1 Introduction

This section presents an analysis for the problem of predicting the surface pressure distribution over a two-dimensional airfoil in a steady compressible potential flow. The flow field under consideration is inviscid, irrotational and the outer free stream velocity is limited to Mach numbers less than one. The main purpose of this section is to develop a numerical procedure that can be applied when the incompressible assumption is not valid.

The solution procedure is similar to the incompressible flow calculation described in the preceding sections. The solution is reached in two steps. The initial pressure distribution is obtained by using the incompressible flow calculation and the solution is converted into the corresponding compressible solution by means of subsequent iterations which take into account the compressibility effect. The procedure is repeated until the solution converges. Details about the calculation method are fully described in the next sections.

4.2 Potential calculation method for compressible flow

The equations are substantially modified to take into account the compressibility factor. However, the calculation procedure remains the same. The problem is divided into an upper and lower plane as shown in Figure 2.2 and the mathematical problem will be solved independantly for the lower and upper plane. The outer flow field is described by the following nonlinear system of equations for the upper plane:

$$\nabla^2 \phi^u = M^2 H^u(\phi) \tag{4.1}$$

$$\left[\frac{\partial \phi^u}{\partial y} \right]_{y=s(x)} = f(x) \tag{4.2}$$

where,

$$H(\phi) = \left[\phi_x^2 \phi_{xx} + 2 \phi_x \phi_y \phi_{xy} + \phi_y^2 \phi_{yy} \right]$$

$$+ \frac{\gamma-1}{2} \left[(\phi_x^2 + \phi_y^2 - 1)(\phi_{xx} + \phi_{yy}) \right] \tag{4.3}$$

On the solid surface of the airfoil the total velocity has to satisfy the tangency condition:

$$\left[\frac{\partial \phi}{\partial y} \right]_{y=s(x)} = \begin{cases} U(x) & -\infty < x < -a \\ \dfrac{\partial \phi}{\partial x} \dfrac{\partial s^u}{\partial x} & -a \le x \le a \\ W(x) & a < x < +\infty \end{cases} \tag{4.4}$$

A similar set of equations can be derived for the lower half

plane.

A successive approximation approach similar to the Rayleigh-Janzen method (2) will be adopted and we will iterate upon the solution for the compressible flow field. In order to apply this procedure, the maximum local Mach number will be restricted to values less than one. Using this approach and denoting each iteration with the superscript (n), equation 4.1 and 4.2 can be rewritten as shown below:

$$\nabla^2 \phi^{u^{(n+1)}} = M^2 H^{u^{(n)}}(x,y) \tag{4.5}$$

$$\left[\frac{\partial \phi}{\partial y} \right]_{y=s(x)} = \begin{cases} U(x) & -\infty < x < -a \\ \dfrac{\partial \phi}{\partial x} \dfrac{\partial s^{u}}{\partial x} & -a \le x \le a \\ W(x) & a < x < +\infty \end{cases} \tag{4.6}$$

The advantage of this approach is that for each approximation, the equations are linear and we can exploit several analytical techniques. For each approximation, we can decompose the potential function into the known freestream value plus the disturbance due to the airfoil. This follows from the linearity of the differential equation, and does not imply a small disturbance approximation.

We have:

$$\phi(x,y) = \phi_{\infty} + \tilde{\phi}(x,y) \tag{4.7}$$

$$U(x) = U_{\infty} + \tilde{U}(x) \tag{4.8}$$

$$W(x) = W_{\infty} + \tilde{W}(x) \tag{4.9}$$

Where $\tilde{U}(x)$ and $\tilde{W}(x)$ represent the unknown upstream and downstream influence of the airfoil. Substituting these expressions into equation 4.5 and 4.6 yields,

$$\nabla^2 \tilde{\phi}^{u^{(n+1)}} = M^2 H^{u^{(n)}}(x,y) \tag{4.10}$$

$$\left[\frac{\partial \tilde{\phi}^{u^{(n)}}}{\partial y} \right]_{y=s(x)} = f^{u^{(n)}}(x) \tag{4.11}$$

$$\tilde{\phi}_x \, , \, \tilde{\phi}_y \to 0 \text{ as } |x| \to \infty \tag{4.12}$$

In equation 5.11, $f^u(x)$ is now given by:

$$f^u(x)=\begin{cases}\tilde{U}(x) & x<-a \\ (\tilde{\phi}^u_x+\phi_{\infty x})\dfrac{ds^u}{dx}-U_\infty & |x|\le a \\ \tilde{W}(x) & x>a\end{cases} \qquad (4.13)$$

It will be advantageous to rewrite $H(x,y)$ as shown below,

$$H(x,y) = H_m(x,y) + H_T(x,y)$$

where,

$$H_m(x,y) = \frac{1}{2}\left(\phi_x\frac{\partial}{\partial x}+\phi_y\frac{\partial}{\partial y}\right)U^2 \qquad (4.14)$$

$$H_T(x,y) = \frac{\gamma-1}{2}\left[(U^2-1)\nabla^2\phi\right] \qquad (4.15)$$

$$U^2=\phi_x^2+\phi_y^2 \qquad (4.16)$$

The system of equations given by 4.10 through 4.16 can now be solved in both the upper and lower planes. At each iteration, the solution is matched along the x-axis for $|x|>a$. This constraint, along with the requirement given below, uniquely determines the flow field,

$$\tilde{U}(x)\to 0 \quad \text{as } |x|\to\infty \qquad (4.17)$$

$$\tilde{W}(x)\to 0 \quad \text{as } |x|\to\infty \qquad (4.18)$$

$$\tilde{W}(+a) = \left[\frac{\partial \phi}{\partial y}\right]_{x=+a} \qquad (4.19)$$

4.3 Iteration procedure for the symmetrical flow case

In order to illustrate the salient feature of the method, the less complicated case of a symmetrical flow will be considered. For this case, the upper and lower problems uncouple, i.e. ,

$$\tilde{U}(x) = \tilde{W}(x) = 0 \qquad (4.20)$$

and we can drop the superscript (u). The basic, n=0, approximation corresponding to the incompressible case is given in chapter 2. The incompressible flow solution serves as the basic, n=0, solution to the compressible flow problem. To compute a second approximation, the following system must be solved.

$$\nabla^2 \tilde{\phi}^{(2)} = M^2 H_m^{(1)}(x,y) \qquad (4.21)$$

$$\left[\frac{\partial \tilde{\phi}^{(2)}}{\partial y}\right]_{y=s(x)} = f^{(2)}(x) \qquad (4.22)$$

$$\tilde{\phi}_x^{(2)} , \tilde{\phi}_y^{(2)} \to 0 \text{ as } |x| \to \infty \qquad (4.23)$$

In equation 4.21 we used the result $\nabla^2 \phi^{(1)} = 0$, to eliminate the

$H_T^{(1)}(x,y)$ term. Taking the Fourier transform and eliminating the $O(\varepsilon^2)$ term, we find the following solution.

$$\frac{\partial \tilde{\phi}^{(2)}}{\partial x} = \frac{\partial \tilde{\phi H}^{(2)}}{\partial x} + \frac{M^2}{2\pi} \int_0^{\infty+\infty} \int_{-\infty} H_M(\xi,\eta) \left[\frac{x-\xi}{r} + \frac{x-\xi}{r} \right] d\xi \, d\eta \tag{4.24}$$

The real advantage of this approach is that the surface integral in equation 4.24 can be simplified to a line integral. This is accomplished as follows: Using the operator form of $H_m(\xi,\eta)$, applying integration by parts, and invoking the Integral Mean Value Theorem to remove v^2 , the surface integral reduces to,

$$\bar{v}^2 \int_0^{\infty+\infty} \int_{-\infty} \nabla . K_D \vec{v} \, d\xi \, d\eta + \int_{-\infty}^{+\infty} \left(\frac{\partial K_D}{\partial x} \frac{\partial \phi}{\partial \eta} v^2 \right)_{y=s(x)} d\xi$$

In this expression, $K_D(x,y;\xi,\eta)$ is the kernel function given in equation 4.24 and $K_x = \partial K_D / \partial x$. The advantage of this re-formulation now becomes apparent. Applying Green's theorem to the surface integral, we see that it becomes zero, and the final result becomes,

$$\frac{\partial \tilde{\phi}^{(2)}}{\partial x} = \frac{\partial \tilde{\phi}^{(1)}}{\partial x} + \frac{1}{\pi}\int_{-a}^{+a} f^{(1)}(\xi)\left[\frac{M^2}{2} u^{(1)\,2}\right]\frac{x-\xi}{(x-\xi)^2+(y-s)^2}\,d\xi \tag{4.25}$$

Equation 4.25 represents a Fredholm integral equation which can be solved for $\partial\phi/\partial x$. The term in brackets represent the first compressibility correction to the basic (incompressible) flow solution.

Successive approximations can now be determined immediately, once the differential system is stated. For example, the third approximation is given by,

$$\nabla^2 \tilde{\phi}^{(3)} = M^2\left[H_m^{(2)}(x,y) + H_T^{(2)}(x,y)\right] \tag{4.26}$$

which becomes,

$$\nabla^2 \tilde{\phi}^{(3)} = M^2 H_m^{(2)} + M^4 \frac{\gamma-1}{2}(u^2-1)^{(2)} H_m^{(1)} \tag{4.27}$$

The associated surface boundary condition is

$$\left[\frac{\partial \tilde{\phi}^{(3)}}{\partial y}\right]_{y=s(x)} = f^{(3)}(x) \tag{4.28}$$

and the solution is given by:

$$\frac{\partial \tilde{\phi}^{(3)}}{\partial x} = \frac{\partial \tilde{\phi}^{(1)}}{\partial x} + \frac{1}{\pi}\int_{-a}^{+a} f^{(2)}(\xi)\left[\frac{M^2}{2}\,U^{(2)^2}\right] K_x(x,y;\xi)\,d\xi$$

$$+\frac{1}{\pi}\int_{-a}^{+a} f^{(1)}(\xi)\left[\frac{M^4}{4}(\gamma-1)\,U^{(1)^2}\right](U^{(2)^2}-1)\,K_x(x,y;\xi)\,d\xi \qquad (4.29)$$

In equation 4.29, $K_x(x,y,\xi)$ is determined by:

$$K_x(x,y;\xi) = \frac{x-\xi}{(x-\xi)^2+(y-s)^2} \qquad (4.30)$$

Higher approximation, i.e., $n \geq 4$, can now be found by inspection, once the differencial equation is written.

4.4 Numerical results

A numerical result is presented for a symmetrical flow over the airfoil NACA 0012. The calculation is validated by comparison with another numerical computation from the computer code developed at NASA Langley [12].

An abbreviated iteration scheme was adapted for solving the compressible flow. This scheme is different from the derived relation given by equation (4.29) which was used in reference [9]. It includes only the first correction term given by equation (4.25), however, an iteration in the computer

program is performed on the integral equation until the solution reaches a converged value. The series of equations used for this approach are shown below for n≤3.

$$\frac{\partial \tilde{\phi}^{(2)}}{\partial x} = \frac{\partial \tilde{\phi}^{(1)}}{\partial x} + \frac{1}{\pi}\int_{-1}^{1} f(\xi)^{(1)} \frac{M_\infty^2}{2} q^{(1)^2} K_x \, d\xi \qquad (4.31)$$

$$\frac{\partial \tilde{\phi}^{(3)}}{\partial x} = \frac{\partial \tilde{\phi}^{(1)}}{\partial x} + \frac{1}{\pi}\int_{-1}^{1} f(\xi)^{(2)} \frac{M_\infty^2}{2} q^{(2)^2} K_x \, d\xi \qquad (4.32)$$

$$\frac{\partial \tilde{\phi}^{(4)}}{\partial x} = \frac{\partial \tilde{\phi}^{(1)}}{\partial x} + \frac{1}{\pi}\int_{-1}^{1} f(\xi)^{(3)} \frac{M_\infty^2}{2} q^{(3)^2} K_x \, d\xi \qquad (4.33)$$

A subroutine was added to the code to compute the compressible flow from the results of the incompressible calculation. Results are presented in terms of the compressible pressure coefficient which is given by:

$$C_p = \frac{2}{\gamma M_\infty^2}\left[\left[1+\frac{\gamma-1}{2a_\infty^2}\left(U_\infty - q^2\right)\right]^{\frac{\gamma}{\gamma-1}} - 1\right] \qquad (4.34)$$

Figure 4.1 shows the numerical solution using a polynomial of degree 20 with equally spaced points for a subsonic flow at M=.6 . The figure demonstrates the effect of the 3 successive iterations and also shows the convergence trend. The first iteration shows a considerable inaccuracy

compared to the NASA code solution and the second and third iterations are in good agreement with the NASA solution. It is found that convergence occurs after 3 iterative calculations and additional iterations do not achieve better convergence. Thirty seconds of computer time on the CDC Dual Cyber were required to produce the result shown in Figure 4.1.

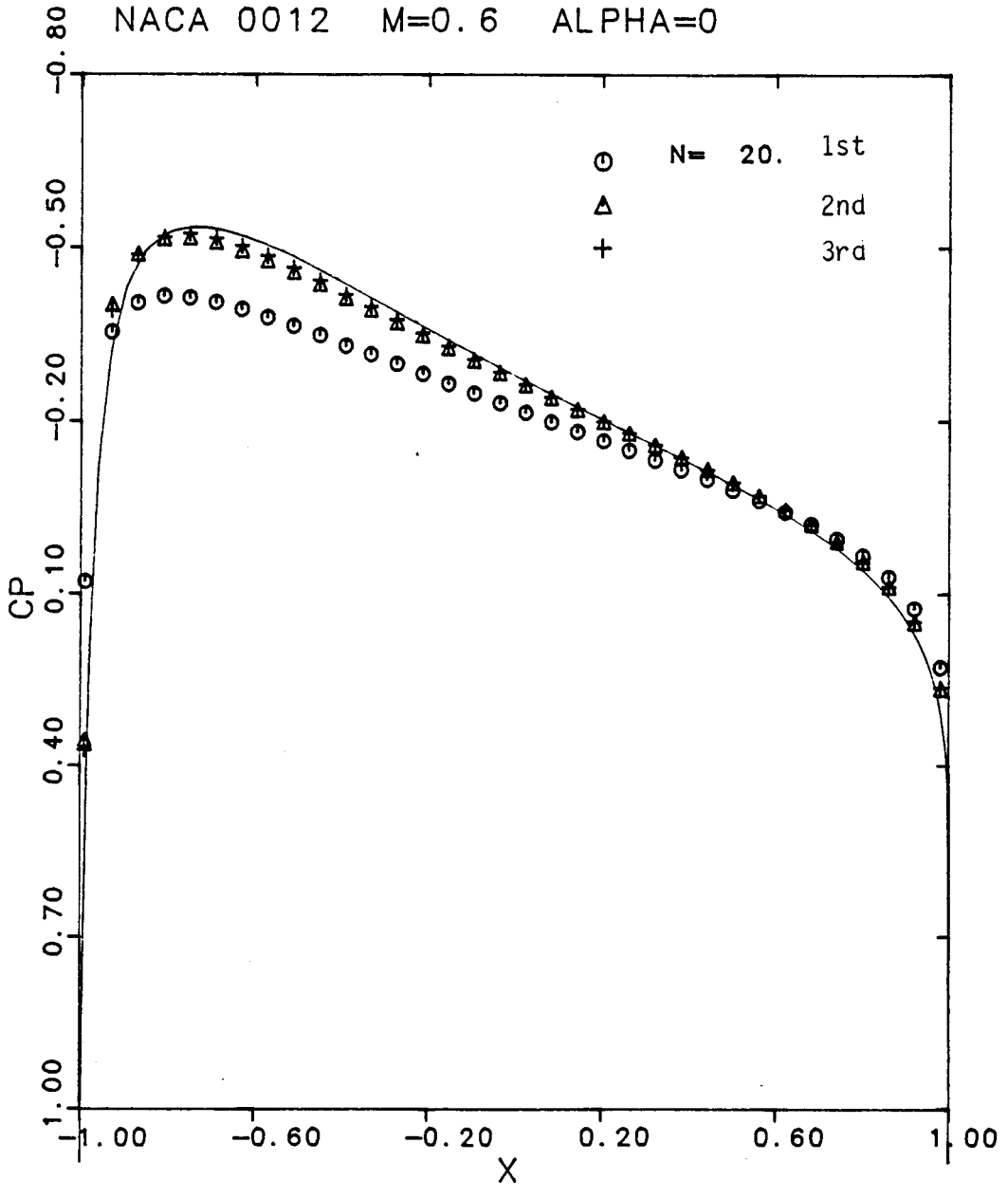

NACA 0012 M=0.6 ALPHA=0
N= 20.
1st
2nd
3rd
CP
-0.80
-0.50
-0.20
0.10
0.40
0.70
1.00
-1.00
-0.60
-0.20
0.20
0.60
1.00
X

Figure 4.1

CHAPTER 5

CONCLUSIONS

Based on the cases examined, the computer programs performed well for two-dimensional airfoils of arbitrary thickness at moderate angle of attack. All the results were very accurate with the exeption of the Joukowski airfoil. The problem may be caused by the presence of the cusped trailing edge which introduces an additional condition. More likely, it is due to the error introduced in the conformal mapping. Hess [15] encountered similar difficulties when calculating the surface pressure distribution for airfoils with very thin trailing edges using a surface singularity distributions method. Hess used an additional parabolic vorticity variation that provided a satisfactory solution for thin trailing-edge airfoils. Using a similar method, Zedan [11] solved the direct and inverse problems of potential flow around an axisymmetric body using an axial source distribution. However, Zedan's method also had difficulty when solving the flow around airfoils with sharp corners or sudden changes in slope.

In this study, the polynomial method provided an efficient and satisfactory solution to two-dimensional flow problems for airfoils with finite trailing edge angles.

The use of double precision variables has proven to be

useful and gives stable and consistent results. Accurate solutions for the surface pressure distribution can be obtained on most airfoils by using 20 to 30 collocation points, especially if the calculation is made using the geometrically increasing grid. A typical case using 24 collocation points requires less than 10 seconds of computer time. One of the most important features of the method is its ability to deal with airfoils of any shapes by adjusting the value of the slope in the subroutine of the main computer program.

APPENDIX A

IDEAL FLOW OVER A JOUKOWSKI AIRFOIL

UPWASH AND DOWNWASH VELOCITY CALCULATION

A-1 Introduction

The upwash and downwash velocities given by the equations 2.31 and 2.32 are obtained by approximating the real airfoil with an equivalent Joukowski airfoil of the same thickness. The solution for the flow about the Joukowski airfoil is accomplished using the traditional transformed plane and then the solution is shifted back into the real plane. The Joukowski method has the advantage that it determines the flow field anywhere in the real plane. In the present appendix we make an extensive study on the y-component of the velocity along the x-axis upstream and downstream of the airfoil as shown in figure A-1.

A-2 Flow about Joukowski airfoil

Referring to figure A-2, we consider the transformation $z=\zeta+c^2/\zeta$ from the transformed plane into the real plane, in which $z=x+iy$ and $\zeta=\xi+i\eta$ are complex variables. The transformation maps the circle of radius r_0 centered at the origin of the ζ plane into a Joukowski airfoil in the physical

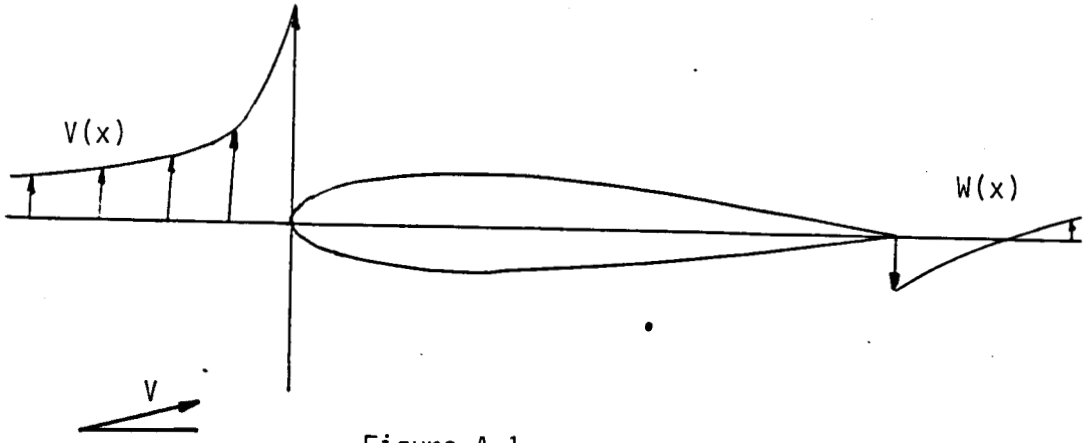


Figure A.1

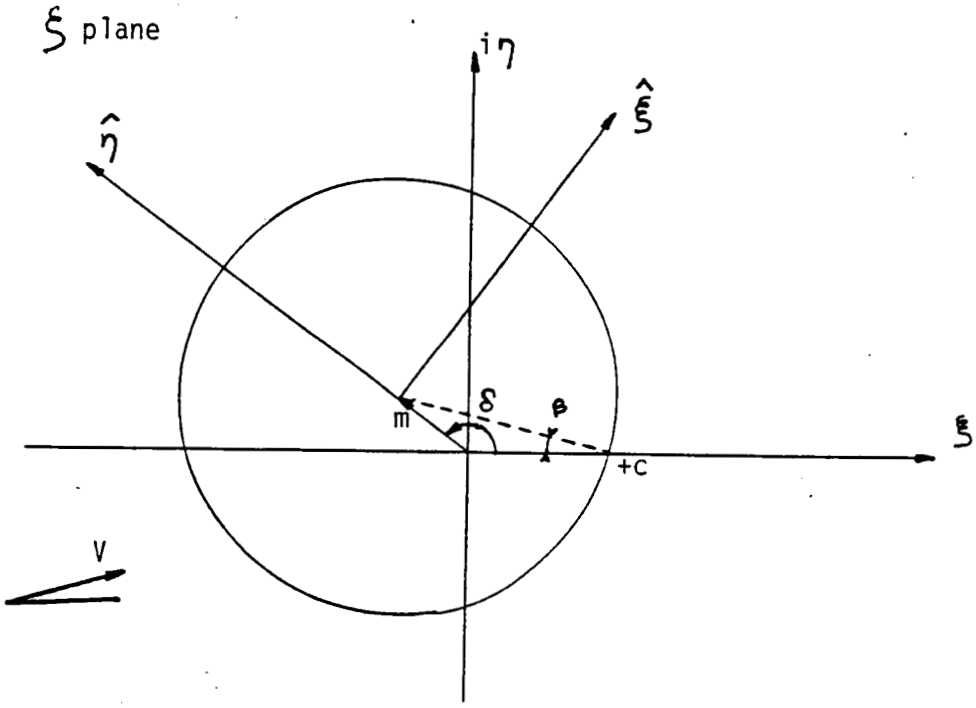


Figure A.2

plane, whose chord is sligtly greater than 2. In particular, the point $\zeta=1$ is mapped into the sharp trailing edge of the airfoil. The shape of the airfoil is controlled by varying the two parameters m and δ. For the present calculation, the airfoil becomes a symmetrical airfoil when $\delta=\pi$. The radius r_0 of the circle and the angle β shown in figure 1 can be expressed in terms of m and δ.

$$r_0 = \sqrt{1-2\,m\cos\delta+m^2} \tag{A.1}$$

$$\beta = \tan^{-1}\left[\frac{m\sin\delta}{1-m\cos\delta}\right] \tag{A.2}$$

These expressions will be used in the later analysis. The varialbies m and δ which describe the displacement of the circle center in the ζ plane are directly related to the camber ratio and to the thickness ratio by the following relations:

$$\frac{m}{c}\sin\delta = 2\,\frac{h}{l} \tag{A.3}$$

$$-\frac{m}{c}\cos\delta = \frac{4}{3\sqrt{3}}\,\frac{t}{l} \tag{A.4}$$

where l is the total length of the chord of the airfoil. Finally we have a complete description of the airfoil parameters with c which describes the position where the circle in the ζ plane cuts the ξ-axis and we note that c is given by:

$$\frac{c}{l} = \frac{1}{4} \tag{A.5}$$

Under the same transformation, a uniform flow in the ζ plane which makes an angle α with the horizontal x-axis, maps into a uniform flow with the same orientation in the physical plane. Let **F** be the complex potential of the flow in the ζ plane, the complex potential consists of a uniform flow about a cylinder with the proper circulation that satisfies the Kutta condition in the real plane. In the current notation, the potential is:

$$F = U\left[\hat{\zeta} + \frac{r_0^2}{\hat{\zeta}}\right] + i\frac{\Gamma_a}{2\pi}\ln\frac{\hat{\zeta}}{r_0} \tag{A.6}$$

where:

$$\zeta = \left[\hat{\zeta} + me^{i\delta}\right]e^{-i\alpha} \tag{A.7}$$

In order to satisfy the Kutta condition, the rear stagnation point needs to be positionned at an angle $\alpha+\beta$ which yields the relation:

$$\sin(\alpha+\beta) = \frac{\Gamma_a}{4\pi r_0 U} \tag{A.8}$$

The complex velocity of the flow about the airfoil is then derived by the equation:

C - 2

$$W(z) = \frac{\hat{dF}}{dz} \tag{A.9}$$

the inverse of the Joukowski transformation is:

$$\zeta = \frac{z}{2} \pm \sqrt{\left(\frac{z}{2}\right)^2 - c^2} \tag{A.10}$$

We also have by definition $W(z)=u-iv$

Inserting A.7 and A.10 into equation A.6, F can be written as:

$$F=\left[\frac{z}{2} \pm \sqrt{\frac{z^2}{4} - \frac{1}{4}} +m \right] e^{-i\alpha} + \frac{r_0^2 e^{i\alpha}}{\left[\frac{z}{2} \pm \sqrt{\frac{z^2}{4} - \frac{1}{4}} +m \right]}$$

$$+ 2 r_0 i \sin\alpha \ln\left[\frac{z}{2} \pm \sqrt{\frac{z^2}{4} - \frac{1}{4}} +m \right] \tag{A.11}$$

After taking the derivative of F and separating the real and imaginary part, the y-component of the velocity along the x-axis upstream and downstream of the airfoil is given by:

$$v(x)= \sqrt{\frac{x-1}{x+1}} \left[\frac{\frac{x}{2} \pm \sqrt{\frac{x^2}{4} - \frac{1}{4}}}{\frac{x}{2} \pm \sqrt{\frac{x^2}{4} - \frac{1}{4}} +m} \right]^2 \sin\alpha \tag{A.12}$$

The mathematical problem for the equivalent Joukowski airfoil is now stated as follows: For a given airfoil with a specified thickness at a given angle of attack α, we have determined an approximation for the y-component of the

velocity upstream and downstream of the airfoil by using an equivalent Joukowski airfoil with same specified thickness and angle of attack. Note that the position of the leading edge of the described Joukowski airfoil is unknown. It may be calculated using the transformation and referring to figure 2, we have:

$$\zeta_{LE} = b\, e^{i\pi} \tag{A.13}$$

The parameter, **b** is given by:

$$\frac{b}{c} = 1 + \frac{m}{c}\left[\cos(\delta-\pi) - \cos\delta\right] \tag{A.14}$$

Using the definition of the transformation yields,

$$z = b\, e^{i\pi} + \frac{c^2}{b\, e^{i\pi}} \tag{A.15}$$

The x-coordinate of the leading edge is obtained by taking the real part of A.15. For example, for a symmetrical Joukowski airfoil with a specified thickness of 12% , we have:

$$x_{LE} = -1.014405264 \tag{A.16}$$

APPENDIX B

AIRFOIL INTEGRALS

The following are the Cauchy principal values for $\kappa^2 < 1$

1. $$\int_{-1}^{1} \frac{1}{\kappa-\xi}\, d\xi = \ln \frac{1+\kappa}{1-\kappa}$$

2. $$\int_{-1}^{1} \frac{\xi}{\kappa-\xi}\, d\xi = \kappa \ln \frac{1+\kappa}{1-\kappa} - 2$$

3. $$\int_{-1}^{1} \frac{\xi^2}{\kappa-\xi}\, d\xi = \kappa \left[\kappa \ln \frac{1+\kappa}{1-\kappa} - 2 \right]$$

4. $$\int_{-1}^{1} \frac{\xi^3}{\kappa-\xi}\, d\xi = \kappa^2 \left[\kappa \ln \frac{1+\kappa}{1-\kappa} - 2 \right] - \frac{2}{3}$$

5. $$\int_{-1}^{1} \frac{\xi^n}{\kappa-\xi}\, d\xi = \kappa \int_{-1}^{1} \frac{\xi^{n-1}}{\kappa-\xi}\, d\xi - \frac{1-(-1)^n}{n}$$

6. $$\int_{-1}^{1} \frac{1}{\sqrt{1-\xi^2}\,(\kappa-\xi)}\, d\xi = 0$$

7. $$\int_{-1}^{1} \frac{\xi}{\sqrt{1-\xi^2}\,(\kappa-\xi)}\, d\xi = -\pi$$

8. $$\int_{-1}^{1} \frac{\xi^2}{\sqrt{1-\xi^2}\,(x-\xi)}\,d\xi = -\pi x$$

9. $$\int_{-1}^{1} \frac{\xi^3}{\sqrt{1-\xi^2}\,(x-\xi)}\,d\xi = -\pi\left[x^2 + \frac{1}{2}\right]$$

10. $$\int_{-1}^{1} \frac{\xi^4}{\sqrt{1-\xi^2}\,(x-\xi)}\,d\xi = -\pi x\left[x^2 + \frac{1}{2}\right]$$

11. $$\int_{-1}^{1} \frac{\xi^5}{\sqrt{1-\xi^2}\,(x-\xi)}\,d\xi = -\pi\left[x^4 + \frac{1}{2}x^2 + \frac{3}{8}\right]$$

12. $$\int_{-1}^{1} \frac{\xi^6}{\sqrt{1-\xi^2}\,(x-\xi)}\,d\xi = -\pi x\left[x^4 + \frac{1}{2}x^2 + \frac{3}{8}\right]$$

13. $$\int_{-1}^{1} \frac{\xi^n}{\sqrt{1-\xi^2}\,(x-\xi)}\,d\xi = x\int_{-1}^{1} \frac{\xi^{n-1}}{\sqrt{1-\xi^2}\,(x-\xi)}\,d\xi -$$

$$\frac{\pi}{2}\left[1-(-1)^n\right]\frac{1\,(3)\ldots(n-2)}{2\,(4)\ldots(n-1)}$$

14. $$\int_{-1}^{1} \frac{\sqrt{1-\xi^2}}{x-\xi}\,d\xi = \pi x$$

15. $$\int_{-1}^{1} \frac{\xi\sqrt{1-\xi^2}}{x-\xi}\,d\xi = \pi\left[x^2 - \frac{1}{2}\right]$$

16. $$\int_{-1}^{1} \frac{\xi^2\sqrt{1-\xi^2}}{x-\xi}\,d\xi = \pi x\left[x^2 - \frac{1}{2}\right]$$

17. $$\int_{-1}^{1} \frac{\xi^3\sqrt{1-\xi^2}}{x-\xi}\,d\xi = \pi\left[x^4 - \frac{1}{2}x^2 - \frac{1}{8}\right]$$

18. $$\int_{-1}^{1} \frac{\sqrt{1+\xi}}{\sqrt{1-\xi}\,(x-\xi)}\,d\xi = -\pi$$

REFERENCES

[1] Munk, M.M. General Theory of thin wing sections . Rep. N.A.C.A. 142, 1922.

[2] Birnbaum, W. Die Tragende Wirbelflache als Hilfmittel zur Behandlung des Ebenen Problems der Tragflugeltheorie . Z. Angew. Math. Mech. 3, 290, 1923.

[3] Glauert, H. The Elements of Aerofoil and Airscrew Theory . 1st Edition, Cambridge University Press, 1926, Cambridge.

[4] Karamcheti, K. Principles of Ideal-Fluid Aerodynamics . John Wiley and Sons, 1980.

[5] Maskew, B. and Woodward, F.A. Symmetrical Singularity Model for Lifting Potential Flow Analysis . Journal of aircraft, Vol.13, September 1976, p733.

[6] Taylor, T.D. Numerical Methods for Predicting Subsonic, Transonic and Supersonic Flow . P.F. Yaggy, 1974.

[7] Hess, J.L. Review of Integral Equation Techniques for Solving Potential-Flow Problems with Emphasis on the Surface-Source Method . Computer Method in Applied Mechanics and Engineering, Vol 5 (1975) 145-196.

[8] Panton, R.L. Incompressible Flow . John Wiley and Sons, 1984.

[9] Wilson, D.E. A New Singular Integral Method for Compressible Potential Flow . AIAA-85-0481.

[10] Marshall, F.J. Design Problem in Hydrodynamics . Journal of Hydronautics, Vol 4, p136.

[11] Zedan, M.F. and Dalton, C. Incompressible, Irrotational, Axisymmetric Flow about a Body of Revolution: the Inverse Problem . Journal of Hydronautics, Vol. 12, Jan. 1978, pp41-46.

[12] Street, C.L., Zang, T.A., Hussaini, M.Y., Spectral Multigrid Methods with Applications to Transonic Potential Flow . ICASE, Report No. 83-11, April 29,1983.

[13] Basu,B.C. ,A Mean Camberline Singularity Method for Two-Dimensional Steady and Oscillatory Aerofoils and Control Surfaces in Inviscid Incompressible Flow . Cp No 1391, October 1976

[14] Currie, I.G. Fundamental Mechanics of Fluids . McGraw-hill, 1974.

[15] Hess, J.L. The Use of Higher-Order Surface Singularity Distributions to Obtain Improved Potential Flow Solutions for Two-Dimensional Lifting Airfoils. Computer Methods in Applied Mechanics and Engineering 5 (1975) 11-35

```
C*****************************************************************
C
C     COMPUTATION OF POTENTIAL FLOW AROUNF TWO-DIMENSIONAL AIRFOILS
C           USING A SINGULAR INTEGRAL METHOD:
C                DIRECT PROBLEM
C
C*****************************************************************
      PROGRAM DBALCP(ANSWER,PLOT,OUTPUT,JOUK0,JOUK4,JOUK10,
     $TAPE2=ANSWER,TAPE8=JOUK0,TAPE9=JOUK4,TAPE10=JOUK10)
C*****************************************************************
C
      DIMENSION X1(205),CPU(205),X2(205),CP2(205),X3(410),CP(410)
     $         ,PHU1(205),CCOM1(205),CCOM2(205),CCOM3(205),CCOM4(205)
     $         ,PHU2(205),QISQ(205)
      DOUBLE PRECISION A(50,51),DETER,ALPHA,PI,X,Z,SX,R
C
C-----------------------------------------------------------------
C
C     INPUT QUANTITIES:
C
C     POLYNOM OF DEGREE N
      N=20
C
C                    AIRFOIL: 1,ELLIPTIC AIRFOIL AT 12% ALPHA=0)
C                             2,NACA0012,ALPHA=0,ICOMP=1 OR 2)
C                             3,NACA0012,ALPHA=4,ICOMP=1)
C                             4,NACA0012,ALPHA=10,ICOMP=1)
C                             5,JOUKOWSKI AIRFOIL,ALPHA=0,JOUK0)
C                             6,JOUKOWSKI AIRFOIL,ALPHA=4,JOUK4)
C                             7,JOUKOWSKI AIRFOIL,ALPHA=10,JOUK10)
C                             8,CYLINDER)
      KAI=2
C
C     IMPLEMENTATION:         1,EQUALLY SPACED POINTS
C                             2,EXPONENTIAL QUADRATURE
      IPOS=2
C
C                  POLYNOM: 1,POWER SERIES OF X
C                           2,SQUARE ROOT AND POWER SERIES OF X
C                           3,AIRFOIL POLYNOMIALS
      IPOLY=1
C
C     ICOMP=1:INCOMPRESSIBLE
C           2:COMPRESSIBLE
      ICOMP=2
C
C     EXPONENTIAL RATIO
      R=1.1D+0
C
C-----------------------------------------------------------------
C
C     ANGLE OF ATTACK
      PI=3.141592654D+00
      ALPHA=0.D+0
      IF(KAI.EQ.3 .OR. KAI.EQ.6) ALPHA=(4.D+0)*PI/(180.D+0)
      IF(KAI.EQ.4 .OR. KAI.EQ.7) ALPHA=(10.D+0)*PI/(180.D+0)
      KA2=1
C
C-----------------------------------------------------------------
C
C     COMPUTES THE MATRIX ELEMENTS ( N EQUATIONS, N UNKNOWNS)
```

```
C
   33 CONTINUE
      I=1
   20 CALL POSIT(X,I,N,IPOS,R)
      CALL MAT1(A,I,X,IPOLY,N)
      CALL DFDX(Z,KAI,X,KA2)
      IF(DABS(Z).LE.(1.D-5)) THEN
         N=N-1
         GO TO 33
      ENDIF
      CALL MAT2(A,N,I,X,IPOLY,KA2,Z)
C
CC-----------------------------------------------------------------
C
C     COMPUTES THE UPWASH AND DOWNWASH VELOCITY INTEGRALS
C
      SX=0.D+0
      IF(ALPHA.EQ.(0.D+0)) GO TO 27
         CALL VELEXT(SX,ALPHA,X)
   27 CONTINUE
      IF(KA2.EQ.1) A(I,N+1)=-SX-DCOS(ALPHA)+DSIN(ALPHA)*
     $             DLOG(((1.D+0)+X)/((1.D+0)-X))/PI
      IF(KA2.EQ.2) A(I,N+1)=-SX+DCOS(ALPHA)+DSIN(ALPHA)*
     $             DLOG(((1.D+0)+X)/((1.D+0)-X))/PI
      I=I+1
      IF(I.LE.N) GO TO 20
C
CC-----------------------------------------------------------------
C
C     GAUSS-JORDAN REDUCTION
C
      DETER=0.D+0
      CALL GAUSS(A,N,DETER)
      WRITE(2,203) DETER
      WRITE(2,208)
      WRITE(2,201) (A(I,N+1),I=1,N)
C
CC-----------------------------------------------------------------
C
C    COMPUTES THE PRESSURE DISTRIBUTION USING THE MATRIX COEFFICIENTS
C     PHU1=X COMPONENT OF THE DISTURBANCE VELOCITY
C
      X1(1)=-1.+2./FLOAT(201)
      DO 36 I=1,200
         X=DBLE(X1(I))
         CALL DFDX(Z,KAI,X,KA2)
         IF(DABS(Z).LE.(1.D-5)) THEN
            CPU(I)=CPU(I-1)
            GO TO 36
         ENDIF
         CALL CPDI(A,Z,N,PHI,X,IPOLY)
         PHU1(I)=PHI-1.
         CPU(I)=1.-(PHI**2)*(1.+SNGL(Z)*SNGL(Z))
      Q1SQ(I)=(PHI**2)*(1.+SNGL(Z)*SNGL(Z))
   36    X1(I+1)=X1(I)+2./FLOAT(201)
   37 CONTINUE
C
C     CALCULATES THE COMPRESSIBILITY FACTOR
C
      IF(ICOMP.EQ.2) CALL CALCOM(X1,PHU1,PHU2,CCOM1,CCOM2,CCOM3,
     $                           CCOM4,KAI,KA2)
      WRITE(2,220)
```

```
      DO 40 I=1,200,5
   40 WRITE(2,221) X1(I),PHU1(I),Q1SQ(I),CCOM1(I),PHU2(I)
  220 FORMAT(/7X,'X',13X,'PH1',5X,'Q1 SQUARE',8X,'CP1 COMP',7X,
     $       'PH2'/)
  221 FORMAT(5(F11.5,3X))
      IF(ALPHA.EQ.(0.D+0)) GO TO 31
      IF(KA2.EQ.2) GO TO 31
C
C-----------------------------------------------------------------
C
C     FOR  A NON ZERO ANGLE OF ATTACK, CALCULATES THE
C     LOWER PLANE DISTRIBUTION
C
      KA2=2
      DO 32 I=1,200
         X2(I)=X1(I)
   32    CP2(I)=CPU(I)
      GO TO 33
   31 CONTINUE
C
C-----------------------------------------------------------------
C
C     READS NUMERICAL PRESSURE DISTRIBUTION FOR COMPARISON
C
      CALL COMP(N2,CP,X3,KAI,ICOMP)
C
  203 FORMAT(/,'DETER =',D14.10/)
  201 FORMAT(1H,13D10.7)
  208 FORMAT(/'POLYNOMIAL COEFFICIENTS'/)
  205 FORMAT(2F10.6)
  206 FORMAT(F14.6)
C
C-----------------------------------------------------------------
C
C     PLOTS THE PRESSURE DISTRIBUTION
C
      IF(ICOMP.EQ.1) THEN
      CALL FIPLOT(X1,CPU,X2,CP2,X3,CP,N2,KAI,ALPHA,IPOS,IPOLY,N,R)
      ELSE
      CALL COMPLOT(X1,CCOM1,CCOM2,CCOM3,CCOM4,X3,CP,N2,KAI,ALPHA,
     $             IPOS,IPOLY,N,R)
      ENDIF
      STOP
      END
C
C*****************************************************************
      SUBROUTINE FIPLOT(X1,CPU,X2,CP2,X3,CP,N2,KAI,ALPHA,IPOS,IPOLY,
     $                  N,R)
C*****************************************************************
C
C     PLOTTING SUBPROGRAM
C
      DIMENSION X1(205),CPU(205),X2(205),CP2(205),X3(410),CP(410)
      DOUBLE PRECISION ALPHA,R
C
      CALL PLOTS(0,0,4LPLOT)
      CALL FACTOR(.95)
C
C-----------------------------------------------------------------
C     SCALES OF THE AXIS
      N1=200
      IF(IPOLY.EQ.3) N1=194
```

```
      IF(KAI.EQ.1 .OR. KAI.EQ.8) N1=100
      X1(N1+1)=-1.
      X1(N1+2)=.4
      IF(KAI.EQ.1 .OR. KAI.EQ.8) X1(N1+2)=.2
      X2(N1+1)=X1(N1+1)
      X2(N1+2)=X1(N1+2)
      X3(N2+1)=X1(N1+1)
      X3(N2+2)=X1(N1+2)
      CPU(N1+1)=1.
      CPU(N1+2)=CP(N2+2)
      CP2(N1+1)=1.
      CP2(N1+2)=CP(N2+2)
      CP(N2+1)=1.
C
C-----------------------------------------------------------------
C     AXIS AND ORIGIN
      CALL AXIS(2.2,2.2,1HX,-1,5.,0.0,X1(N1+1),X1(N1+2))
      CALL AXIS(2.2,2.2,2HCP,2,6.0,90.0,CPU(N1+1),CPU(N1+2))
      CALL AXIS(2.2,8.2,1H ,-1,5.,0.,999.,1.)
      CALL AXIS(7.2,2.2,0,0,6.,90.,999.,1.)
      CALL ORIGIN(2.2,2.2,0)
C
C-----------------------------------------------------------------
C     DESIGNATION OF THE CURVES
      CALL PLOT(.5,5.75,3)
      CALL PLOT(.8,5.75,2)
      IF(KAI.EQ.1) CALL SYMBOL(1.,5.75,.1,'EXACT ANALYTICAL',0.,16)
      IF(KAI.EQ.6 .OR. KAI.EQ.6 .OR. KAI.EQ.7) THEN
         CALL SYMBOL(1.,5.75,.1,'THEORETICAL',0.,11)
      ENDIF
      IF(KAI.EQ.8) CALL SYMBOL(1.,5.75,.1,'THEORETICAL',0.,11)
      IF(KAI.EQ.2 .OR. KAI.EQ.3 ) THEN
         CALL SYMBOL(1.,5.75,.1,'NUMERICAL TAB [12]',0.,18)
      ENDIF
      IF(KAI.EQ.4) CALL SYMBOL(1.,5.75,.1,'NUMERICAL DATA[13]',
     $0.,18)
C
      CALL LINE(X3,CP,N2,1,0,2)
      CALL LINE(X1,CPU,N1,1,-6,1)
      IF(ALPHA.NE.(0.D+00)) CALL LINE(X2,CP2,N1,1,-6,1)
C
      CALL SYMBOL(.6,5.5,.1,1,0.,-1)
      CALL SYMBOL(1.,5.5,.1,'POLYNOMIAL N=',0.,13)
      CALL NUMBER(2.5,5.5,.1,FLOAT(N),0.,0)
      IF(IPOS.EQ.2) THEN
         CALL SYMBOL(4.,5.75,.1,'R=',0.,2)
         CALL NUMBER(4.3,5.75,.1,SNGL(R),0.,1)
      ENDIF
      IF(IPOLY.EQ.2) CALL SYMBOL(4.,5.5,.1,'POLY=2',0.,6)
      IF(IPOLY.EQ.3) CALL SYMBOL(4.,5.5,.1,'POLY=3',0.,6)
C
C-----------------------------------------------------------------
C     TITLE
      IF(KAI.EQ.1) CALL SYMBOL(.2,6.25,.14,'ELLIPSE  ALPHA=0',0.,16)
      IF(KAI.EQ.8) CALL SYMBOL(.2,6.25,.14,'CYLINDER',0.,8)
      IF(KAI.EQ.2) CALL SYMBOL(.2,6.25,.14
     $,'NACA 0012 AIRFOIL ALPHA=0',0.,25)
      IF(KAI.EQ.3) CALL SYMBOL(.2,6.25,.14
     $,'NACA 0012 AIRFOIL ALPHA=4',0.,25)
      IF(KAI.EQ.4) CALL SYMBOL(.2,6.25,.14
     $,'NACA 0012 AIRFOIL ALPHA=10',0.,26)
      IF(KAI.EQ.5) CALL SYMBOL(.2,6.25,.14
```

```
     $,'JOUKOWSKI AIRFOIL ALPHA=0',0.,25)
      IF(KAI.EQ.6) CALL SYMBOL(.2,6.25,.14
     $,'JOUKOWSKI AIRFOIL ALPHA=4',0.,25)
      IF(KAI.EQ.7) CALL SYMBOL(.2,6.25,.14
     $,'JOUKOWSKI AIRFOIL ALPHA=10',0.,26)
      CALL PLOT(-.5,-.5,999)
    7 CONTINUE
      RETURN
      END
C
C*************************************************************************
      SUBROUTINE DFDX(Z,KAI,X,KA2)
C*************************************************************************
C
C     SUBPROGRAM THAT CALCULATES THE DERIVATIVE OF THE AIRFOIL
C
      DOUBLE PRECISION Z,X
C
C-------------------------------------------------------------------------
C     ELLIPTIC AIRFOIL
      IF(KAI.EQ.1) Z=-(.12D+0)*X/DSQRT((1.D+0)-X*X)
C
C     CYLINDER
      IF(KAI.EQ.8) Z=-X/DSQRT((1.D+0)-X*X)
C
C     NACA0012
      IF(KAI.EQ.2 .OR. KAI.EQ.3 .OR. KAI.EQ.4) THEN
      Z=(1.2D+0)*((.074225D+0)/DSQRT((X+(1.D+0))/
     $     (2.D+0))-(.063D+0)-(.3516D+0)*((X+(1.D+0))/(2.D+0))
     $     +(.42645D+0)*(((X+(1.D+0))/(2.D+0))**(2.D+0))-(.203D+0)*
     $     (((X+(1.D+0))/(2.D+0))**(3.D+0)))
      ENDIF
C
C     JOUKOWSKI AIRFOIL
      IF(KAI.EQ.5 .OR. KAI.EQ.6 .OR. KAI.EQ.7) THEN
      Z=((4.D+0)/((3.D+0)*DSQRT(3.D+0)))*(.12D+0)*
     $   (-DSQRT((1.D+0)-X*X)-((1.D+0)-X)*X/DSQRT((1.D+0)-X*X))
      ENDIF
C
      IF(KA2.EQ.2) Z=-Z
      RETURN
      END
C
C*************************************************************************
      SUBROUTINE VELEXT(SX,AL2,X)
C*************************************************************************
C
C     CALCULATES THE UPWASH AND DOWNWASH VELOCITIES
C
      DOUBLE PRECISION VEL,XL,AL2,RM,XSLE,SX,X,PI
      PI=3.14159265D+00
C     RM=          COEFFICIENT FOR THE UPWASH AND DOWNWASH JOUKOWSKY
C                  CALCULATION OF THE EXTERNAL VELOCITY
C     RM=SQRT(THICKNESS*THICKNESS/27+CAMBER*CAMBER/4)
C
      RM=.0461880 2D+0
      XSLE=-1.01440526 4D+0
      XL=-(1.D+0)-(.02D+0)/(2.D+0)
C
C-------------------------------------------------------------------------
C     UPWASH CALCULATION
      DO 25 IL=1,500
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
      XS=(1.D+0)-((1.D+0)-XL)*((1.D+0)-XSLE)/(2.D+0)
      VEL=(DSQRT((XS-(1.D+0))/(XS+(1.D+0)))*(((XS/(2.D+0)-
     $DSQRT(XS*XS/(4.D+0)-(.25D+0)))/(XS/(2.D+0)-DSQRT(XS*XS/
     $(4.D+0)-(.25D+0))+RM))**(2.D+0))-(1.D+0))*DSIN(AL2)
        SX=SX+VEL*(.02D+0)/(X-XL)/PI
   25 XL=XL-(.02D+0)
C
C-----------------------------------------------------------------------
C     DOWNWASH CALCULATION
      XL=(1.D+0)+(.02D+0)/(2.D+0)
      DO 26 IL=1,500
      XS=(1.D+0)-((1.D+0)-XL)*((1.D+0)-XSLE)/(2.D+0)
      VEL=(DSQRT((XS-(1.D+0))/(XS+(1.D+0)))*(((XS/(2.D+0)+
     $DSQRT(XS*XS/(4.D+0)-(.25D+0)))/(XS/(2.D+0)+DSQRT(XS*XS/
     $(4.D+0)-(.25D+0))+RM))**(2.D+0))-(1.D+0))*DSIN(AL2)
        SX=SX+VEL*(.02D+0)/(X-XL)/PI
   26 XL=XL+(.02D+0)
      RETURN
      END
C
C***********************************************************************
      SUBROUTINE GAUSS(A,N,DETER)
C***********************************************************************
C
C     GAUSS-JORDAN REDUCTION
C     THIS SUBPROGRAM FINDS THE SOLUTION VECTOR CORRESPONDING TO A
C     SET OF N SIMULTANEOUS LINEAR EQUATIONS USING THE GAUSS-JORDAN
C     REDUCTION ALGORITHM WITH THE DIAGONAL PIVOT STRATEGY.
C
      DOUBLE PRECISION A(50,51),DETER,EPS
      EPS=1.D-10
      NPLUSM=N+1
C     ..... BEGIN ELIMINATION PROCEDURE .....
      DETER=(1.D+0)
      DO 9 K=1,N
C     ..... UPDATE THE DETERMINANT VALUE .....
      DETER=DETER*A(K,K)
C     ..... CHECK FOR PIVOT ELEMENT TOO SMALL .....
      IF(DABS(A(K,K)).GT.EPS) GO TO 5
      GO TO 7
C     ..... NORMALIZE THE PIVOT ROW .....
    5 KP1=K+1
      DO 6 J=KP1,NPLUSM
    6 A(K,J)=A(K,J)/A(K,K)
      A(K,K)=1.D+0
C     ..... ELIMINATE K(TH) COLUMN ELEMENTS EXECPT FOR PIVOT .....
      DO 11 I=1,N
      IF( I.EQ.K .OR. A(I,K).EQ.(0.D+0) ) GO TO 11
      DO 8 J=KP1,NPLUSM
    8 A(I,J)=A(I,J)-A(I,K)*A(K,J)
      A(I,K)=0.D+0
   11 CONTINUE
    9 CONTINUE
      GO TO 1
    7 CONTINUE
      WRITE(2,202)
  202 FORMAT(/,'SMALL PIVOT - MATRIX MAY BE SINGULAR')
      STOP
    1 CONTINUE
      RETURN
      END
C
```

```
C*********************************************************************
      SUBROUTINE POSIT(X,I,N,IPOS,R)
C*********************************************************************
C
C     SUBPROGRAM THAT IMPLEMENTS THE LOCATION OF THE SET OF QUADRATURE
C     POINTS
C
C
C     LIST OF THE VARIABLES:
C     XP      LOCATION OF THE POINTS
C     N       NUMBER OF POINTS
C     IPOS    TYPE OF SPACING
C     R       EXPONENTIAL RATIO
C
      DOUBLE PRECISION XP(50),R,DELTA,DX,XL,X
      IF(IPOS.EQ.1) THEN
         XP(1)=-(1.D+00)+(2.D+00)/DBLE(FLOAT(N+1))
         DO 10 J=2,N
         XP(J)=XP(J-1)+(2.D+00)/DBLE(FLOAT(N+1))
   10    CONTINUE
      ENDIF
C
      IF(IPOS.EQ.2) THEN
         NTE=N/2
         NLE=N-NTE
         DELTA=(1.D+0)/(((1.D+0)-(R**NLE))/((1.D+0)-R))
         XL=-1.D+0
         DO 1 J=1,NLE
         DX=DELTA*(R**(J-1))
         XP(J)=XL+DX/(2.D+0)
    1    XL=XL+DX
C
         DELTA=(1.D+0)/(((1.D+0)-(R**NTE))/((1.D+0)-R))
         XL=1.D+0
         DO 2 J=1,NTE
         DX=DELTA*(R**(J-1))
         XP(N+1-J)=XL-DX/(2.D+0)
    2    XL=XL-DX
      ENDIF
      X=XP(I)
      RETURN
      END
C
C*********************************************************************
      SUBROUTINE MAT1(A,I,X,IPOLY,N)
C*********************************************************************
      DOUBLE PRECISION A(50,51),X,RMT,PI,RM1
C
      PI=3.141592654D+0
C
C     SUBPROGRAM THAT COMPUTES THE MATRIX ELEMENTS
C
      IF(IPOLY.EQ.1) THEN
         A(I,1)=(DLOG(((1.D+0)+X)/((1.D+0)-X)))/PI
         DO 1 J=2,N
         A(I,J)=X*A(I,J-1)-((1.D+00)-DBLE(FLOAT((-1)**(J-1))))/
     $   (DBLE(FLOAT(J-1))*PI)
    1    CONTINUE
      ENDIF
C
      IF(IPOLY.EQ.2) THEN
         A(I,1)=X
```

```
      A(I,2)=X*X-(.5D+0)
      A(I,3)=X*A(I,2)
      A(I,4)=X*A(I,3)-(1.D+0)/(8.D+0)
      A(I,5)=X*A(I,4)
      DO 2 J=6,N
         RMT=1.D+0
         DO 3 IMT=1,J-3,2
    3    RMT=RMT*DBLE(FLOAT(IMT/(IMT+1)))
      A(I,J)=X*A(I,J-1)-(.5D+0)*DBLE(FLOAT(1-((-1)**(J-1))))
     $ *RMT/DBLE(FLOAT(J))
    2 CONTINUE
      ENDIF
C
      IF(IPOLY.EQ.3) THEN
         A(I,1)=1.D+0
         A(I,2)=(1.D+0)+X
         A(I,3)=(.5D+0)+X+X*X
         A(I,4)=(.5D+0)+(.5D+0)*X+X*X+(X**3)
         DO 4 J=5,N
            RMT=1.D+0
            RM1=1.D+0
            DO 5 IMT=1,J-3,2
    5          RMT=RMT*DBLE(FLOAT(IMT/(IMT+1)))
            DO 6 IMT=1,J-2,2
    6          RM1=RM1*DBLE(FLOAT(IMT/(IMT+1)))
            A(I,J)=X*A(I,J-1)+(.5D+0)*DBLE(FLOAT(1-((-1)**(J-1))))
     $      *RMT+(.5D+0)*DBLE(FLOAT(1-((-1)**(J))))*RM1
    4       CONTINUE
         ENDIF
C
      RETURN
      END
C
C
C*****************************************************************
      SUBROUTINE MAT2(A,N,I,X,IPOLY,KA2,Z)
C*****************************************************************
      DOUBLE PRECISION A(50,51),X,Z,RS
C
      RS=1.D+0
      IF(KA2.EQ.2) RS=-1.D+0
C
      IF(IPOLY.EQ.1) THEN
         A(I,1)=A(I,1)-RS/Z
         DO 1 J=2,N
         A(I,J)=A(I,J)-RS*(X**(J-1))/Z
    1    CONTINUE
      ENDIF
C
      IF(IPOLY.EQ.2) THEN
         A(I,1)=A(I,1)-RS*DSQRT((1.D+0)-X*X)/Z
         DO 2 J=2,N
         A(I,J)=A(I,J)-RS*DSQRT((1.D+0)-X*X)*(X**(J-1))/Z
    2    CONTINUE
      ENDIF
C
      IF(IPOLY.EQ.3) THEN
         A(I,1)=A(I,1)+RS*DSQRT(((1.D+0)+X)/((1.D+0)-X))/Z
         DO 3 J=2,N
         A(I,J)=A(I,J)+RS*DSQRT(((1.D+0)+X)/((1.D+0)-X))*(X**(J-1))/Z
    3    CONTINUE
      ENDIF
```

```
C
      RETURN
      END
C
C*****************************************************************
      SUBROUTINE CPDI(A,Z,N,PHI,X,IPOLY)
C*****************************************************************
C     PRESSURE DISTRIBUTION
C
      DOUBLE PRECISION A(50,51),Z,X
C
      IF(IPOLY.EQ.1) THEN
         PHI=SNGL(A(1,N+1))/SNGL(Z)
         DO  1 J=2,N
            PHI=PHI+SNGL(A(J,N+1))*SNGL(X**(J-1))/SNGL(Z)
    1    CONTINUE
      ENDIF
      IF(IPOLY.EQ.2) THEN
         PHI=SQRT(1.-SNGL(X*X))*SNGL(A(1,N+1))/SNGL(Z)
         DO 2 J=2,N
         PHI=PHI+SQRT(1.-SNGL(X*X))*SNGL(A(J,N+1))*SNGL(X**(J-1))/
     $   SNGL(Z)
    2 CONTINUE
      ENDIF
C
      IF(IPOLY.EQ.3) THEN
         PHI=-SQRT((1.+SNGL(X))/(1.-SNGL(X)))*SNGL(A(1,N+1))/SNGL(Z)
         DO 3 J=2,N
         PHI=PHI-SQRT((1.+SNGL(X))/(1.-SNGL(X)))*SNGL(A(J,N+1))*
     $   SNGL(X**(J-1))/SNGL(Z)
    3    CONTINUE
      ENDIF
C
      RETURN
      END
C
C*****************************************************************
      SUBROUTINE COMP(N2,CP,X3,KAI,ICOMP)
C*****************************************************************
C
C     READ THE CP DISTRIBUTION FOR COMPARISON
C
      DIMENSION CP(410),X3(410),XP1(50),CP1(50)
     $         ,XP2(40),CP2(40),XP4(40),CP4(40)
     $         ,XP3(70),CP3(70)
C
C     DATA FOR NACA 0012 ALPHA=0
      DATA XP1/1.000000, .997865, .991852, .982340, .969515,
     $          .953555, .934629, .912939, .888573, .861806,
     $          .832804, .801769, .768917, .734470, .698658,
     $          .661720, .623898, .585439, .546589, .507593,
     $          .468694, .430128, .392122, .354896, .318658,
     $          .283605, .249925, .217791, .187366, .158804,
     $          .132244, .107823, .085651, .065848, .048511,
     $          .033727, .021571, .012098, .005346, .001327,
     $          .000000/
C
      DATA CP1/1.000000, .358876, .262520, .203777, .158001,
     $          .119665, .086223, .056269, .028921, .003559,
     $         -.020291,-.043022,-.064972,-.086434,-.107666,
     $         -.128882,-.150250,-.171887,-.193848,-.216129,
     $         -.238655,-.261284,-.283805,-.305942,-.327360,
```

```
$          -.347662,-.366402,-.383074,-.397099,-.407794,
$          -.414288,-.415372,-.409194,-.392672,-.360298,
$          -.301726,-.196984,-.038853, .315239, .755544,
$          1.000000/
C
C      DATA FOR NACA 0012 ALPHA=0 COMPRESSIBLE FLOW M=0.6
       DATA XP2/1.000000, .996731, .987540, .973043, .953575,
$           .929474, .901076, .868734, .832822, .793736,
$           .751896, .707743, .661736, .614348, .566061,
$           .517355, .468706, .420576, .373408, .327623,
$           .283614, .241747, .202362, .165768, .132250,
$           .102067, .075452, .052614, .033731, .018953,
$           .008384, .002081, .000000/
C
       DATA CP2/1.093269, .387102, .283197, .209395, .151856,
$           .102060, .058212, .017938,-.019380,-.055012,
$          -.089464,-.123615,-.157880,-.192841,-.228690,
$          -.265637,-.303488,-.341979,-.380389,-.417905,
$          -.453192,-.484843,-.510824,-.528978,-.536172,
$          -.528654,-.500269,-.441061,-.330939,-.131477,
$           .222871, .756856,1.093269/
C
C      DATA FOR NACA 0012 ALPHA=4
       DATA XP3/1.000000,.996731,.987540,.973043,.953575,
$           .929474,.901076,.868734,.832822,.793736,
$           .751896,.707743,.661736,.614348,.566061,
$           .517355,.468706,.420576,.373408,.327623,
$           .283614,.241747,.202362,.165768,.132250,
$           .102067,.075452,.052614,.033731,.018953,
$           .008384,.002081,.000000,.002081,.008384,
$           .018953,.033731,.052614,.075452,.102067,
$           .132250,.165768,.202362,.241747,.283614,
$           .327623,.373408,.420576,.468706,.517355,
$           .566061,.614348,.661736,.707743,.751896,
$           .793736,.832822,.868734,.901076,.929474,
$           .953575,.973043,.987540,.996731/
C
       DATA CP3/1.064172,  .374256,  .280179,  .218264,  .171865,
$           .134179,  .102582,  .075460,  .051872,  .030885,
$           .012101, -.005338, -.021403, -.036427, -.050363,
$          -.063049, -.074006, -.082602, -.087860, -.088644,
$          -.083448, -.070609, -.048907, -.013241,  .036878,
$           .106406,  .201227,  .329655,  .503111,  .730817,
$           .981974, 1.021016,  .305786, -.951149,-1.685110,
$         -1.813038,-1.693127,-1.531778,-1.381723,-1.250222,
$         -1.134866,-1.029778, -.934240, -.845727, -.762944,
$          -.685359, -.612491, -.544249, -.480336, -.420631,
$          -.364722, -.312287, -.262711, -.215467, -.169755,
$          -.124925, -.080040, -.034382,  .013201,  .063638,
$           .118939,  .181396,  .257856,  .364680/
C
C      DATA FOR NACA 0012 ALPHA=10
       DATA XP4/-.9944,-.9890,-.9833,-.9690,-.9444,
$          -.9000,-.8000,-.7000,-.6000,-.4000,
$          -.2000, .0000, .2000, .4000, .6000,
$           .8000, .8611, .9778, .8611, .8000,
$           .6000, .4000, .2000, .0000,-.2000,
$          -.4000,-.6000,-.8000,-.8500,-.9443,
$          -.9722,-.9800,-.9910/
C
       DATA CP4/-6.1100,-6.4166,-6.0000,-5.0000,-4.0000,
$          -3.0000,-2.1670,-1.7222,-1.4400,-1.0000,
```

```
     $          -.7500, -.5670, -.4440, -.3055, -.1940,
     $          -.0555,  .0000,  .2778,  .1667,  .1667,
     $           .1550,  .1667,  .1667,  .2220,  .2500,
     $           .3060,  .4170,  .6670,  .7780, 1.0000,
     $           .5000,  .0000,-1.0000/
C
C      (AIRFOIL NACA12 AT 4 DEGREES OF ANGLE OF ATTACK)
       IF(KAI.EQ.3) THEN
       N2=64
       CP(66)=-.5
       DO 41 I=1,64
       X3(I)=XP3(I)
       CP(I)=CP3(I)
       CP(I)=CP(I)*SQRT(1.-.5*.5)
    41 X3(I)=2.*(X3(I)-.5)
       ENDIF
C
C      (AIRFOIL NACA0012 AT 10 DEGREES OF ANGLE OF ATTACK)
       IF(KAI.EQ.4) THEN
       N2=33
       CP(35)=-1.
       DO 48 I=1,33
       X3(I)=XP4(I)
       CP(I)=CP4(I)
    48 CONTINUE
       ENDIF
C
C      (AIRFOIL NACA12 AT 0 DEGREE OF ANGLE OF ATTACK)
       IF(KAI.EQ.2 .AND. ICOMP.EQ.1) THEN
       N2=41
       CP(N2+2)=-.3
       DO 44 I=1,N2
       X3(I)=XP1(I)
       CP(I)=CP1(I)
    44 X3(I)=2.*(X3(I)-.5)
       ENDIF
C
       IF(KAI.EQ.2 .AND. ICOMP.EQ.2) THEN
       N2=33
       CP(N2+2)=-.3
       DO 45 I=1,N2
       X3(I)=XP2(I)
       CP(I)=CP2(I)
    45 X3(I)=2.*(X3(I)-.5)
       ENDIF
C
C      (ELLIPTIC AIRFOIL)
       IF(KAI.EQ.1) THEN
       N2=101
       CP(103)=-.3
       X3(1)=-1.
       CP(1)=1.
       DO 46 I=2,101
       X3(I)=X3(I-1)+1./100.
       QE=(1.+.12)/SQRT(1.+.12*.12*X3(I)*X3(I)/(1.-X3(I)*X3(I)))
    46 CP(I)=1.-QE*QE
       ENDIF
C
C      (CYLINDER)
       IF(KAI.EQ.8) THEN
       N2=101
       CP(103)=-.8
```

```
      X3(1)=-1.
      CP(1)=1.
      DO 40 I=2,101
      X3(I)=X3(I-1)+1./100.
   40 CP(I)=-3.+4.*X3(I)*X3(I)
      ENDIF
C
C     (JOUKOWSKI AIRFOIL AT 2 DEGREE OF ANGLE OF ATTACK)
      IF(KAI.EQ.5) THEN
      N2=202
      CP(N2+2)=-.3
      DO 60 I=1,202
   60 READ(8,205) X3(I),CP(I)
      ENDIF
C
C     (JOUKOWSKI AIRFOIL AT 4 DEGREES OF ANGLE OF ATTACK)
      IF(KAI.EQ.6) THEN
      N2=404
      CP(N2+2)=-.5
      DO 61 I=1,202
   61 READ(9,205) X3(I),CP(I)
      DO 62 I=1,202
   62 READ(9,205) X3(405-I),CP(405-I)
      ENDIF
C
C     (JOUKOWSKI AIRFOIL AT 10 DEGREES OF ANGLE OF ATTACK)
      IF(KAI.EQ.7)
      N2=404
      CP(N2+2)=-1.
      DO 63 I=1,202
   63 READ(10,205) X3(I),CP(I)
      DO 64 I=1,202
   64 READ(10,205) X3(405-I),CP(405-I)
      ENDIF
C
  205 FORMAT(2F10.6)
      RETURN
      END
C
C*********************************************************************
      SUBROUTINE CALCOM(X1,PHU1,PHU2,CCOM1,CCOM2,CCOM3,CCOM4,KAI,KA2)
C*********************************************************************
C
C     CALCULATION OF THE COMPRESSIBILITY FACTOR
C
      DIMENSION X1(205),PHU1(205),PHU2(205),PHU3(205),PHU4(205)
      DIMENSION CCOM1(205),CCOM2(205),CCOM3(205),CCOM4(205)
      DOUBLE PRECISION A(50,51),DETER,Z
      N1=200
      PI=3.141592654
C
C     PHU1 =X-COMPONENT OF THE DISTURBANCE VELOCITY
C     CCOM1=PRESSURE COEFFICIENT
C---------------------------------------------------------------------
C
C     CALCULATES CP(1ST)
      CALL CALCPC(X1,PHU1,CCOM1)
C
C     CALCULATES CP(2ND)
      DO 1 I=1,200
      PHU2(I)=PHU1(I)
C     CALCULATES THE POLYNOMIAL THAT FITS BETWEEN -1 AND +1
```

```
      CALL MATCOM(A,X1,PHU1)
      CALL GAUSS(A,20,DETER)
      RINT=ALOG((1.+X1(I))/(1.-X1(I)))/PI
      PHU2(I)=PHU2(I)+SNGL(A(1,21))*RINT
      DO 2 IM=2,20
      RINT=RINT*X1(I)-(1.-FLOAT((-1)**(IM-1)))
     $     /(FLOAT(IM-1)*PI)
    2 PHU2(I)=PHU2(I)+SNGL(A(IM,21))*RINT
    1 CONTINUE
      CALL CALCPC(X1,PHU2,CCOM2)
C
C     CALCULATES CP(3RD)
      DO 3 I=1,200
      PHU3(I)=PHU1(I)
C     CALCULATES THE POLYNOMIAL THAT FITS BETWEEN -1 AND +1
      CALL MATCOM(A,X1,PHU2)
      CALL GAUSS(A,20,DETER)
      RINT=ALOG((1.+X1(I))/(1.-X1(I)))/PI
      PHU3(I)=PHU3(I)+SNGL(A(1,21))*RINT
      DO 4 IM=2,20
      RINT=RINT*X1(I)-(1.-FLOAT((-1)**(IM-1)))
     $     /(FLOAT(IM-1)*PI)
    4 PHU3(I)=PHU3(I)+SNGL(A(IM,21))*RINT
    3 CONTINUE
      CALL CALCPC(X1,PHU3,CCOM3)
      GO TO 7
C
C     CALCULATES CP(4TH)
      DO 5 I=1,200
      PHU4(I)=PHU1(I)
C     CALCULATES THE POLYNOMIAL THAT FITS BETWEEN -1 AND +1
      CALL MATCOM(A,X1,PHU3)
      CALL GAUSS(A,20,DETER)
      RINT=ALOG((1.+X1(I))/(1.-X1(I)))/PI
      PHU4(I)=PHU4(I)+SNGL(A(1,21))*RINT
      DO 6 IM=2,20
      RINT=RINT*X1(I)-(1.-FLOAT((-1)**(IM-1)))
    6 PHU4(I)=PHU4(I)+SNGL(A(IM,21))*RINT
    5 CONTINUE
      CALL CALCPC(X1,PHU4,CCOM4)
    7 CONTINUE
      RETURN
      END
C
C*****************************************************************
      SUBROUTINE CALCPC(X1,PHU1,CCOM1)
C*****************************************************************
C
C     CALCULATES CP FROM PHI
      DIMENSION X1(205),PHU1(205),CCOM1(205)
      DOUBLE PRECISION Z,X
C
C     PHU1=X-COMPONENT OF THE DISTURBANCE VELOCITY
C
      N1=200
      KA1=2
      KA2=1
      DO 1 I=1,200
         X=DBLE(X1(I))
         CALL DFDX(Z,KA1,X,KA2)
         IF(DABS(Z).LE.(1.D-5)) THEN
            CCOM1(I)=CCOM1(I-1)
```

```
            GO TO 2
         ENDIF
         Q2=((1.+PHU1(I))**2)*(1.+SNGL(Z)*SNGL(Z))
      RGAM=1.4/(1.4-1.)
         RFAC=1.+(1.4-1.)*.3*.6*(1.-Q2)
      GO TO 20
         CCOM1(I)=(1.-Q2)/(SQRT(1.-.6*.6)+.6*.6*(.5-Q2/2.)/
     $                (1.+SQRT(1.-.6*.6)))
   20 CCOM1(I)=(2./(1.4*.6*.6))*((RFAC**RGAM)-1.)
    2 CONTINUE
    1 CONTINUE
      RETURN
      END
C
C*********************************************************************
      SUBROUTINE MATCOM(A,X1,PHU1)
C*********************************************************************
C
C     MATRIX CALCULATION
C
      DIMENSION PHU1(205),X1(205)
      DOUBLE PRECISION A(50,51),XK,Z
C
C     POLYNOMIAL CALCULATION
      RHK=-1.+2./21.
      DO 5 IK=1,20
         XK=DBLE(RHK)
         CALL CORR(RHK,PHK,X1,PHU1,200)
         CALL DFDX(Z,2,XK,1)
         A(IK,1)=(1.D+0)
         DO 6 J=2,20
    6       A(IK,J)=(XK**(J-1))
         A(IK,21)=((1.D+0)+DBLE(PHK))*Z*((.6D+0))*((.6D+0)/(2.D+0))*
     $            ((1.D+0)+DBLE(PHK))*((1.D+0)+DBLE(PHK))*
     $            ((1.D+0)+Z*Z)
    5    RHK=RHK+2./21.
      RETURN
      END
C
C*********************************************************************
      SUBROUTINE COMPLOT(X1,CCOM1,CCOM2,CCOM3,CCOM4,X3,CP,N2,
     $                   KAI,ALPHA,IPOS,IPOLY,N,R)
C*********************************************************************
C
C     PLOTTING ROUTINE
C
      DIMENSION X1(205),CCOM1(205),CCOM2(205),CCOM3(205),CCOM4(205)
      DIMENSION X3(205),CP(205)
      DOUBLE PRECISION ALPHA,R
C
      CALL PLOTS(0,0,4LPLOT)
      CALL FACTOR(.95)
      N1=200
      X1(N1+1)=-1.
      X1(N1+2)=.4
      X3(N2+1)=-1.
      X3(N2+2)=.4
      CP(N2+1)=1.
      CP(N2+2)=-.3
      CCOM1(N1+1)=1.
      CCOM1(N1+2)=CP(N2+2)
      CCOM2(N1+1)=CP(N2+1)
```

```
      CCOM2(N1+2)=CP(N2+2)
      CCOM3(N1+1)=CP(N2+1)
      CCOM3(N1+2)=CP(N2+2)
      CCOM4(N1+1)=CP(N2+1)
      CCOM4(N1+2)=CP(N2+2)
C
C-----------------------------------------------------------------
C     AXIS AND ORIGIN
      CALL AXIS(2.2,2.2,1HX,-1,5.,0.0,X1(N1+1),X1(N1+2))
      CALL AXIS(2.2,2.2,2HCP,2,6.0,90.0,CCOM1(N1+1),CCOM1(N1+2))
      CALL AXIS(2.2,8.2,1H ,-1,5.,0.,999.,1.)
      CALL AXIS(7.2,2.2,0.0,6.,90.,999.,1.)
      CALL ORIGIN(2.2,2.2,0)
      CALL LINE(X1,CCOM1,N1,1,-6,1)
      CALL SYMBOL(3.,5.5,.1,1,0.,-1)
      CALL SYMBOL(3.4,5.5,.1,'N=',0.,2)
      CALL NUMBER(3.8,5.5,.1,FLOAT(N),0.,0)
C
C-----------------------------------------------------------------
      CALL LINE(X1,CCOM2,N1,1,-6,2)
      CALL SYMBOL(3.,5.25,.1,2,0.,-1)
      CALL LINE(X1,CCOM3,N1,1,-6,3)
      CALL SYMBOL(3.,5.,.1,3,0.,-1)
      GO TO 7
      CALL LINE(X1,CCOM4,N1,1,-6,4)
      CALL SYMBOL(3.,4.75,.1,4,0.,-1)
    7 CONTINUE
C
C-----------------------------------------------------------------
C     COMPARISON
      CALL LINE(X3,CP,N2,1,0,2)
C
C-----------------------------------------------------------------
C     TITLE
      GO TO 20
      CALL SYMBOL(1.,.5,.1,'KARMAN-TSIEN CORRECTION FORMULA',
     $0.,31)
   20 CONTINUE
      IF(KAI.EQ.2) CALL SYMBOL(.2,6.25,.14
     $,'NACA 0012  M=0.6  ALPHA=0',0.,25)
      CALL PLOT(-.5,-.5,999)
      RETURN
      END
C
C*****************************************************************
      SUBROUTINE CORR(X,UXS,X1,UX,N1)
C*****************************************************************
C
C     CALCULATES UXS BY INTERPOLATION USING UX(205)
C
      DIMENSION X1(205),UX(205)
C
      IF(X.LE.X1(3)) THEN
      UXS=(X-X1(2))*(X-X1(3))*(X-X1(4))*UX(1)/(X1(1)-X1(2))
     $/(X1(1)-X1(3))/(X1(1)-X1(4))+
     $(X-X1(1))*(X-X1(3))*(X-X1(4))*UX(2)/(X1(2)-X1(1))/(X1(2)-X1(3))
     $/(X1(2)-X1(4))+
     $(X-X1(1))*(X-X1(2))*(X-X1(4))*UX(3)/(X1(3)-X1(1))/(X1(3)-X1(2))
     $/(X1(3)-X1(4))+
     $(X-X1(1))*(X-X1(2))*(X-X1(3))*UX(4)/(X1(4)-X1(1))/(X1(4)-X1(2))
     $/(X1(4)-X1(3))
      ENDIF
```

```
C
      DO 1 I=3,N1-3
      IF(X.GT.X1(I) .AND. X.LE.X1(I+1)) THEN
      UXS=(X-X1(I))*(X-X1(I+1))*(X-X1(I+2))*UX(I-1)/((X1(I-1)-X1(I))*
     $(X1(I-1)-X1(I+1))*(X1(I-1)-X1(I+2)))+
     $(X-X1(I-1))*(X-X1(I+1))*(X-X1(I+2))*UX(I)/((X1(I)-X1(I-1))*
     $(X1(I)-X1(I+1))*(X1(I)-X1(I+2)))+
     $(X-X1(I-1))*(X-X1(I))*(X-X1(I+2))*UX(I+1)/((X1(I+1)-X1(I-1))*
     $(X1(I+1)-X1(I))*(X1(I+1)-X1(I+2)))+
     $(X-X1(I-1))*(X-X1(I))*(X-X1(I+1))*UX(I+2)/((X1(I+2)-X1(I-1))*
     $(X1(I+2)-X1(I))*(X1(I+2)-X1(I+1)))
      ENDIF
    1 CONTINUE
C
      IF(X.GT.X1(N1-2)) THEN
      I=N1-2
      UXS=(X-X1(I))*(X-X1(I+1))*(X-X1(I+2))*UX(I-1)/((X1(I-1)-X1(I))*
     $(X1(I-1)-X1(I+1))*(X1(I-1)-X1(I+2)))+
     $(X-X1(I-1))*(X-X1(I+1))*(X-X1(I+2))*UX(I)/((X1(I)-X1(I-1))*
     $(X1(I)-X1(I+1))*(X1(I)-X1(I+2)))+
     $(X-X1(I-1))*(X-X1(I))*(X-X1(I+2))*UX(I+1)/((X1(I+1)-X1(I-1))*
     $(X1(I+1)-X1(I))*(X1(I+1)-X1(I+2)))+
     $(X-X1(I-1))*(X-X1(I))*(X-X1(I+1))*UX(I+2)/((X1(I+2)-X1(I-1))*
     $(X1(I+2)-X1(I))*(X1(I+2)-X1(I+1)))
      ENDIF
C
      RETURN
      END
```

```
C*****************************************************************
C
C     SOLUTION OF THE INVERSE DESIGN PROBLEM
C
C*****************************************************************
C
      PROGRAM REV4(ANSWER,OUTPUT,COEFPO,CPDATA,CPTAB4,JOUKO,PLOT,
     $TAPE2=ANSWER,TAPE3=COEFPO,TAPE5=CPDATA,TAPE6=CPTAB4,TAPE7=JOUKO)
      DIMENSION XA(50),X1(205),CP(205),UX(205),Y1(205),Y2(205)
      DIMENSION XD1(205),VX(205),U(205)
      DOUBLE PRECISION X,A(50,51),DETER
C
C*****************************************************************
C
C     INPUT QUANTITIES
C
C     POLYNOM OF DEGREE N
      N=24
C     PRESSURE DISTRIBUTION
C        1,THEORETICAL ELLIPTIC  DISTRIBUTION,ALPHA=0)
C        3,POLYNOMIAL NACA0012 (ALCP, COEFPO N=19),ALPHA=0)
C        4,ELLIPTIC POLYNOMIAL (ALCP, COEFPO N=30),ALPHA=0)
C        5,NACA12 DISTRIBUTION FROM TAB,(CPDATA),ALPHA=0)
C        6,NACA12 DISTRIBUTION FROM TAB,(CPTAB4),ALPHA=4)
C        7,CALCULATED JOUKOWSKY DISTRIBUTION (JOUKO),ALPHA=0)
C        8,CYLINDER,ALPHA=0)
      KAI=5
C              DISTURBANCE FIELD: 1,UX
C                                 2,VX
      KAP=2
C
C*****************************************************************
C
C     COMPUTES OR READS THE X-DISTURBANCE VELOCITY FIELD
C
C     THEORETICAL ELLIPTIC DISTRIBUTION
C     THICKNESS RATIO
      RT=.12
      IF(KAI.EQ.1) THEN
         N1=200
         X1(1)=-1.+2./FLOAT(201)
         DO 21 I=1,N1
            UX(I)=(1.+RT)/(1.+RT*RT+X1(I)*X1(I)/(1.-X1(I)*X1(I)))-1.
            VX(I)=-(UX(I)+1.)*RT*X1(I)/SQRT(1.-X1(I)*X1(I))
   21       X1(I+1)=X1(I)+2./FLOAT(201)
      ENDIF
C
C     CYLINDER
      IF(KAI.EQ.8) THEN
         N1=200
         X1(1)=-1.+2./FLOAT(201)
         DO 31 I=1,N1
            UX(I)=2.*(1.-X1(I)*X1(I))-1.
            VX(I)=-2.*X1(I)*SQRT(1.-X1(I)*X1(I))
   31       X1(I+1)=X1(I)+2./FLOAT(201)
      ENDIF
C
C     DATA OF CP FROM PROGRAM: ALCP
```

```
      IF(KAI.EQ.3 .OR. KAI.EQ.4) THEN
         N1=200
         DO 52 I=1,200
            READ(3,205) X1(I),CP(I)
            XS=X1(I)
            CALL DFDX(Z,KAI,XS,RT)
            UX(I)=SQRT((1.-CP(I))/(1+Z*Z))-1.
            VX(I)=(UX(I)+1.)*Z
   52       CONTINUE
      ENDIF

C     FILE OF CP (TAB) AT 0 DEGREE
      IF(KAI.EQ.5) THEN
         N1=40
         DO 23 I=1,N1
            READ(5,205) X1(41-I),CP(41-I)
            X1(41-I)=2.*(X1(41-I)-.5)
            XS=X1(41-I)
            CALL DFDX(Z,KAI,XS,RT)
            UX(41-I)=SQRT((1.-CP(41-I))/(1.+Z*Z))-1.
   23       VX(41-I)=(UX(41-I)+1.)*Z
      ENDIF

C     JOUKOWSKY AIRFOIL
      IF(KAI.EQ.7) THEN
         N1=200
         READ(7,205) X1(1),CP(1)
         DO 57 I=1,200
            READ(7,205) X1(I),CP(I)
            XS=X1(I)
            CALL DFDX(Z,KAI,XS,RT)
            UX(I)=SQRT((1.-CP(I))/(1.+Z*Z))-1.
            VX(I)=(UX(I)+1.)*Z
   57       CONTINUE
      ENDIF
C
CC
C*****************************************************************************
C
C
C     COMPUTES THE MATRIX
C
      X=-(1.D+0)+(2.D+0)/DBLE(FLOAT(N+1))
      DO 26 I=1,N
         XS=SNGL(X)
         CALL CORR(XS,VXS,X1,VX,N1)
         CALL CORR(XS,UXS,X1,UX,N1)
         CALL MATR1(A,I,X,UXS,VXS,N,KAP)
         X=X+(2.D+0)/DBLE(FLOAT(N+1))
   26 CONTINUE
C
CC
C*****************************************************************************
C
C
C     SOLVES FOR THE MATRIX A(N,NPLUSM)
C
      CALL GAUSS(A,N,DETER)
      WRITE(2,203) DETER,N,N+1
      WRITE(2,208)
      DO 5 I=1,N
         XA(I)=SNGL(A(I,N+1))
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
      WRITE(2,201) XA(I)
    5 CONTINUE
C
C
C*****************************************************************
C
C
C     COMPUTES THE AIRFOIL PROFILE USING THE POLYNOMIAL COEFFICIENTS
C
      PI=3.141592654
      IF(KAI.EQ.2 .OR. KAI.EQ.3 .OR. KAI.EQ.5 .OR. KAI.EQ.6
     $ .OR. KAI.EQ.7) THEN
      ND=201
      Y1(1)=0.
      XD1(1)=1.
      DO 28 I=2,201
         XD1(I)=XD1(I-1)-2./FLOAT(201)
         XS=XD1(I)
         IF(KAP.EQ.2) CALL CORR(XS,VXS,X1,VX,N1)
         IF(KAP.EQ.1) CALL CORR(XS,UXS,X1,UX,N1)
         CALL CPDR1(XS,XA,UXS,VXS,N,KAP)
         FX=VXS/(1.+UXS)
         Y1(I)=Y1(I-1)+FX*(XD1(I)-XD1(I-1))
   28    CONTINUE
      ENDIF
C
C     ELLIPTIC AIRFOIL
      IF(KAI.EQ.1 .OR. KAI.EQ.4 .OR. KAI.EQ.8) THEN
      ND=100
      XD1(1)=-1./FLOAT(200)
      Y1(1)=.12
      IF(KAI.EQ.8) Y1(1)=1.
      DO 60 I=1,99
         XD1(I+1)=XD1(I)-2./FLOAT(200)
         XS=XD1(I)
         IF(KAP.EQ.1) CALL CORR(XS,UXS,X1,UX,N1)
         IF(KAP.EQ.2) CALL CORR(XS,VXS,X1,VX,N1)
         CALL CPDR1(XS,XA,UXS,VXS,N,KAP)
         FX=VXS/(1.+UXS)
         Y1(I+1)=Y1(I)+FX*(XD1(I+1)-XD1(I))
   60 CONTINUE
C
      ENDIF
C
C*****************************************************************
C
C     COMPUTATION OF THE THEORITICAL PROFILE FOR COMPARISON
C
      CALL THEO(ND,KAI,XD1,Y2,RT)
C
  202 FORMAT(/,'SMALL PIVOT - MATRIX MAY BE SINGULAR')
  203 FORMAT(/,'DETER = ',D16.12/,'N      = ',I2/,'NPLUSM= ',I2/)
  201 FORMAT(1H,13F10.5)
  208 FORMAT(/'POLYNOMIAL COEFFICIENTS'/)
  205 FORMAT(2F10.6)
  216 FORMAT(I3)
C
C
C*****************************************************************
C
C
C     PLOT OF THE PROFILE OF THE WING
```

```
C
      CALL PLOTS(0,0,4LPLOT)
      CALL FACTOR(.95)
      CALL SCALE(XD1,5.,ND,1)
      XD1(ND+1)=-1.
      XD1(ND+2)=.4
      IF(KAI.EQ.1 .OR. KAI.EQ.4 .OR. KAI.EQ.8) XD1(ND+2)=.2
      CALL SCALE(Y1,6.,ND,1)
      Y1(ND+1)=0.0
      Y1(ND+2)=.03
      IF(KAI.EQ.8) Y1(ND+2)=.2
      Y2(ND+1)=Y1(ND+1)
      Y2(ND+2)=Y1(ND+2)
      CALL AXIS(2.2,2.2,1HX,-1,5.,0.,XD1(ND+1),XD1(ND+2))
      CALL AXIS(2.2,2.2,2HY ,2,6.,90.,Y1(ND+1),Y1(ND+2))
      CALL AXIS(2.2,8.2,0,0,5.,0.,999.,1.)
      CALL AXIS(7.2,2.2,0,0,6.,90.,999.,1.)
      CALL ORIGIN(2.2,2.2,0)
      CALL LINE(XD1,Y1,ND,1,-6,1)
      CALL LINE(XD1,Y2,ND,1,0,2)
      CALL SYMBOL(.6,5.5,.1,1,0.,-1)
      CALL SYMBOL(1.,5.5,.1,'POLYNOMIAL N=',0.,13)
      CALL NUMBER(2.5,5.5,.1,FLOAT(N),0.,0)
      IF(KAI.EQ.3) CALL SYMBOL(.2,6.25,.14,'DATA FROM THE PROGRAM ALCP',
     $0.,26)
      IF(KAI.EQ.4) CALL SYMBOL(.2,6.25,.14,'DATA FROM THE PROGRAM ALCP',
     $0.,26)
      IF(KAI.EQ.5) CALL SYMBOL(.2,6.25,.14,'DATA FROM TAB (0 DEGREE)',
     $0.,24)
      IF(KAI.EQ.1) CALL SYMBOL(.2,6.25,.14,'ELLIPTIC AIRFOIL',0.,16)
      IF(KAI.EQ.8) CALL SYMBOL(.2,6.25,.14,'CYLINDER',0.,8)
      IF(KAI.EQ.7) CALL SYMBOL(.2,6.25,.14,'JOUKOWSKY AIRFOIL',0.,17)
      CALL PLOT(-.5,-.5,999)
7     CONTINUE
      STOP
      END
C
C
C**********************************************************************
C
C
C     CALCULATES THE AIRFOIL DERIVATIVE
C
      SUBROUTINE DFDX(Z,KAI,X,RT)
C     ELLIPTIC AIRFOIL
      IF(KAI.EQ.1 .OR. KAI.EQ.4) Z=-RT*X/(SQRT(1.-X*X))
C     CYLINDER
      IF(KAI.EQ.8) Z=-X/SQRT(1.-X*X)
C     NACA0012
      IF(KAI.EQ.3 .OR. KAI.EQ.5 .OR. KAI.EQ.6) THEN
         Z=10.*RT*(.074225/(SQRT((X+1.)/2.))-.063
     $ -.3516*((X+1.)/2.)+.42645*(((X+1.)/2.)**2.)
     $ -.203*(((X+1.)/2.)**3.))
      ENDIF
C     JOUKOWSKY AIRFOIL
      IF(KAI.EQ.7) Z=-4.*RT*(SQRT(1.-X*X)+(1.-X)*X/SQRT(1.-X*X))/3./
     $SQRT(3.)
      RETURN
      END
C
C
C**********************************************************************
```

```
C
C     CALCULATES UXS BY INTERPOLATION USING UX(205)
C
      SUBROUTINE CORR(X,UXS,X1,UX,N1)
      DIMENSION X1(205),UX(205)
      IF(X.LE.X1(3)) THEN
      UXS=(X-X1(2))*(X-X1(3))*(X-X1(4))*UX(1)/(X1(1)-X1(2))
     $/(X1(1)-X1(3))/(X1(1)-X1(4))+
     $(X-X1(1))*(X-X1(3))*(X-X1(4))*UX(2)/(X1(2)-X1(1))/(X1(2)-X1(3))
     $/(X1(2)-X1(4))+
     $(X-X1(1))*(X-X1(2))*(X-X1(4))*UX(3)/(X1(3)-X1(1))/(X1(3)-X1(2))
     $/(X1(3)-X1(4))+
     $(X-X1(1))*(X-X1(2))*(X-X1(3))*UX(4)/(X1(4)-X1(1))/(X1(4)-X1(2))
     $/(X1(4)-X1(3))
      ENDIF
      DO 1 I=3,N1-3
      IF(X.GT.X1(I) .AND. X.LE.X1(I+1)) THEN
      UXS=(X-X1(I))*(X-X1(I+1))*(X-X1(I+2))*UX(I-1)/((X1(I-1)-X1(I))*
     $(X1(I-1)-X1(I+1))*(X1(I-1)-X1(I+2)))+
     $(X-X1(I-1))*(X-X1(I+1))*(X-X1(I+2))*UX(I)/((X1(I)-X1(I-1))*
     $(X1(I)-X1(I+1))*(X1(I)-X1(I+2)))+
     $(X-X1(I-1))*(X-X1(I))*(X-X1(I+2))*UX(I+1)/((X1(I+1)-X1(I-1))*
     $(X1(I+1)-X1(I))*(X1(I+1)-X1(I+2)))+
     $(X-X1(I-1))*(X-X1(I))*(X-X1(I+1))*UX(I+2)/((X1(I+2)-X1(I-1))*
     $(X1(I+2)-X1(I))*(X1(I+2)-X1(I+1)))
      ENDIF
    1 CONTINUE
      IF(X.GT.X1(N1-2)) THEN
      I=N1-2
      UXS=(X-X1(I))*(X-X1(I+1))*(X-X1(I+2))*UX(I-1)/((X1(I-1)-X1(I))*
     $(X1(I-1)-X1(I+1))*(X1(I-1)-X1(I+2)))+
     $(X-X1(I-1))*(X-X1(I+1))*(X-X1(I+2))*UX(I)/((X1(I)-X1(I-1))*
     $(X1(I)-X1(I+1))*(X1(I)-X1(I+2)))+
     $(X-X1(I-1))*(X-X1(I))*(X-X1(I+2))*UX(I+1)/((X1(I+1)-X1(I-1))*
     $(X1(I+1)-X1(I))*(X1(I+1)-X1(I+2)))+
     $(X-X1(I-1))*(X-X1(I))*(X-X1(I+1))*UX(I+2)/((X1(I+2)-X1(I-1))*
     $(X1(I+2)-X1(I))*(X1(I+2)-X1(I+1)))
      ENDIF
      RETURN
      END
C
C
C**********************************************************************
C
C
C     CALCULATES THE THEORITICAL PROFILE OF THE AIRFOIL
C
      SUBROUTINE THEO(N1,KAI,X1,Y2,RT)
      DIMENSION X1(205),Y2(205)
C
C     ELLIPTIC AIRFOIL
      IF(KAI.EQ.1 .OR. KAI.EQ.4) THEN
         DO 1 I=1,N1
    1    Y2(I)=RT*SQRT(1.-X1(I)*X1(I))
      ENDIF
C
C     CYLINDER
      IF(KAI.EQ.8) THEN
         DO 10 I=1,N1
   10    Y2(I)=SQRT(1.-X1(I)*X1(I))
      ENDIF
C
```

```
C       NACA2012
        IF(KAI.EQ.3 .OR. KAI.EQ.5 .OR. KAI.EQ.6) THEN
           DO 2 I=1,N1
           Y2(I)=10.*RT*(.2969*SQRT(X1(I)*.5+.5)-.126*(X1(I)*.5+.5)
     C     -.3516*((X1(I)*.5+.5)**2)+.2843*((X1(I)*.5+.5)**3)
     C     -.10150*((X1(I)*.5+.5)**4))
       2   CONTINUE
        ENDIF
C
C       JOUKWSKY AIRFOIL                                    ORIGINAL PAGE IS
        IF(KAI.EQ.7) THEN                                   OF POOR QUALITY
           DO 4 I=1,N1
           Y2(I)=4.*RT*(1.-X1(I))*(SQRT(1.-X1(I)*X1(I)))/3./SQRT(3.)
       4   CONTINUE
        ENDIF
        RETURN
        END
C
C
C*********************************************************************
C
C
C       GAUSS-JORDAN REDUCTION
C       THIS SUBPROGRAM FINDS THE SOLUTION VECTOR CORRESPONDING TO A
C       SET OF N SIMULTANEOUS LINEAR EQUATIONS USING THE GAUSS-JORDAN
C       REDUCTION ALGORITHM WITH THE DIAGONAL PIVOT STRATEGY.
C
        SUBROUTINE GAUSS(A,N,DETER)
        DOUBLE PRECISION A(50,51),DETER,EPS
        EPS=1.D-10
        NPLUSM=N+1
C       ..... BEGIN ELIMINATION PROCEDURE .....
        DETER=(1.D+0)
        DO 9 K=1,N
C       ..... UPDATE THE DETERMINANT VALUE .....
        DETER=DETER*A(K,K)
C       ..... CHECK FOR PIVOT ELEMENT TOO SMALL .....
        IF(DABS(A(K,K)).GT.EPS) GO TO 5
        GO TO 7
C       ..... NORMALIZE THE PIVOT ROW .....
       5 KP1=K+1
        DO 6 J=KP1,NPLUSM
       6 A(K,J)=A(K,J)/A(K,K)
        A(K,K)=1.D+0
C       ..... ELIMINATE K(TH) COLUMN ELEMENTS EXECPT FOR PIVOT .....
        DO 11 I=1,N
        IF( I.EQ.K .OR. A(I,K).EQ.(0.D+0) ) GO TO 11
        DO 8 J=KP1,NPLUSM
       8 A(I,J)=A(I,J)-A(I,K)*A(K,J)
        A(I,K)=0.D+0
      11 CONTINUE
       9 CONTINUE
        GO TO 1
       7 CONTINUE
        STOP
       1 CONTINUE
        RETURN
        END
C
C*********************************************************************
C
C       COMPUTES THE MATRIX ELEMENTS
```

```
C
C
      SUBROUTINE MATR1(A,I,X,UXS,VXS,N,KAP)
      DOUBLE PRECISION X,A(50,51),PI
      PI=3.141592654D+0
C
      IF(KAP.EQ.1) THEN
         A(I,1)=(DLOG(((1.D+0)+X)/((1.D+0)-X)))/PI
         DO 1 J=2,N
         A(I,J)=X*A(I,J-1)-((1.D+00)-DBLE(FLOAT((-1)**(J-1))))/
     $   (DBLE(FLOAT(J-1))*PI)
    1    CONTINUE
      A(I,N+1)=DBLE(UXS)
      ENDIF
C
      IF(KAP.EQ.2) THEN
         A(I,1)=1.D+0
         DO 2 J=2,N
         A(I,J)=X**(J-1)
    2    CONTINUE
      A(I,N+1)=DBLE(VXS)
      ENDIF
      RETURN
      END
C
C     PRESSURE DISTRIBUTION
C
      SUBROUTINE CPDR1(XS,XA,UXS,VXS,N,KAP)
      DIMENSION XA(51)
C
      IF(KAP.EQ.1) THEN
      VXS=XA(1)
      DO 1 I=2,N
         VXS=VXS+XA(I)*(XS**(I-1))
    1    CONTINUE
      ENDIF
C
      IF(KAP.EQ.2) THEN
         PI=3.141592654D+0
         RINT=ALOG((1.+XS)/(1.-XS))/PI
         UXS=XA(1)*RINT
         DO 2 J=2,N
            RINT=XS*RINT-(1.-FLOAT((-1)**(J-1)))/(FLOAT(J-1))/PI
            UXS=UXS+XA(J)*RINT
    2    CONTINUE
      ENDIF
C
      RETURN
      END
```