Numerical Solutions to the Small Disturbance Equations

Stanley M. Porembski[1]
Arizona State University Ira. A. Fulton Schools of Engineering, Tempe, AZ, 85281, USA

Analytical solutions for both the subsonic and supersonic flow regimes have been developed through the application of multiple assumptions. These solutions however are unable to account for the transonic regime as both subsonic and supersonic flows exist on the airfoil and some assumptions used in the derivations of the analytical solutions no longer apply. To approximate the aerodynamics of complex airfoils in the transonic flow regime numerical solutions are necessary. A numerical solution to the Prandtl-Glauert equation was created and tested on a symmetrical, circular-arc airfoil at zero angle of attack in both subsonic and transonic conditions. The pressure distribution along with the drag coefficient from the numerical solution were then compared against empirical data to determine the applicability of the Prandtl-Glauert equation on complex airfoil. It was determined that the Prandtl-Glauert equation can accurately simulate the aerodynamics of a thin, complex airfoil in subsonic conditions, but not in transonic conditions.

I. Nomenclature

a	=	Speed of sound
c_p	=	Pressure coefficient
c_d	=	Drag Coefficient
K	=	Transonic similarity parameter
M	=	Mach number
U	=	Velocity
x	=	x coordinate
z	=	z coordinate
Z	=	airfoil surface height
Z'	=	airfoil surface slope
γ	=	Specific heat ratio of air
φ	=	Velocity potential
τ	=	Thickness per unit chord length

II. Introduction

The transonic region is defined as the region of Mach numbers between 0.8 and 1.2 and is classified by the presence of both subsonic and supersonic flow along an airfoil. As the free-stream Mach approaches the transonic region it will eventually reach a point where a Mach of 1 first appears on the airfoil and is called the free-stream Mach number is called the critical Mach number. As the free-stream increases beyond the critical Mach number towards Mach 1, the region of supersonic flow on the surface of the airfoil increases. For the supersonic flow to return to subsonic flow it must go through a shock wave. This shock wave greatly increases the pressure and temperature of the air traveling through it which causes an increase in the drag on the airfoil, called wave drag [5].

This flow regime is typically avoided by supersonic and hypersonic vehicles while accelerating or decelerating through Mach 1 due to the effects of traveling through the region such as the drag-divergence phenomena, the rapid change in center of pressure, and the unpredictability of the shockwaves on control surfaces [4]. The drag-divergence

[1] Arizona State University Undergraduate Student, Aerospace Engineering (Autonomous Vehicle Systems).

phenomenon is where slight changes in Mach number will result in dramatic changes in the drag coefficient when approaching Mach 1 as seen in Fig. 1.

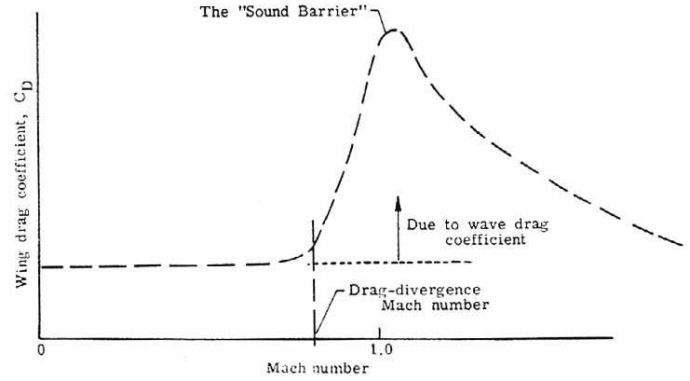


Fig. 1 Variation of an airplane wing drag coefficient with Mach number [5].

This increased drag compared to the subsonic and supersonic regions is caused by the unpredictable shock formations on an airfoil and other general flow instabilities.

This issue with the transonic region is that the linearized equations for the supersonic and subsonic pressure coefficient are not valid close to Mach 1 as these equations predict a pressure singularity of infinite pressure at Mach 1 which does not happen in real life.

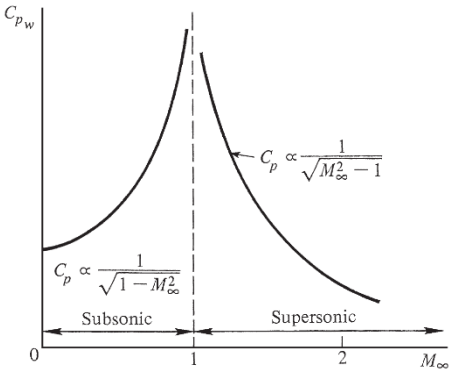


Fig. 2 Variation of the linearized pressure coefficient with Mach number [4].

Because of the non-linearity of the transonic aerodynamics, numerical solutions are utilized as there is no analytical solution. Subsonic flow does not experience many of the issues present in the transonic region, but a numerical solution is commonly used to solve the subsonic aerodynamics of complex boundary conditions. A model used in the numerical solutions is the 2-dimensional velocity perturbation equation which is derived from the conservation of mass and energy.

$$\left[1 - \frac{1}{a^2}\varphi_x^2\right]\varphi_{xx} + \left[1 - \frac{1}{a^2}\varphi_y^2\right]\varphi_{yy} - \frac{2}{a}\varphi_x\varphi_z\varphi_{xy} = 0 \tag{1}$$

This equation is non-linear and must be linearized to be solved. This is accomplished with the assumption that the flow is subsonic over the entire airfoil, inviscid, isentropic, irrotational, compressible, and that the airfoil is at small angles of attack. Defining linear perturbations is also necessary to linearize the velocity perturbation equation and are represented using hats as seen in eq. 2 and eq. 3.

$$\varphi_x = U_\infty + \hat{\varphi}_x \tag{2}$$

$$\varphi_y = \hat{\varphi}_y \tag{3}$$

The simplifications and linear perturbations were used to simplify eq. 1 to the Prandtl-Glauert equation as seen in eq. 4.

$$(1 - M_\infty^2)\hat{\varphi}_{xx} + \hat{\varphi}_{yy} = 0 \tag{4}$$

Using the Prandtl-Glauert equation, the pressure coefficient along the surface of a symmetrical circular-arc airfoil at zero angle of attack using a finite-difference solution. The profile of the airfoil was defined as:

$$Z = \pm 2\tau\left[\frac{1}{4} - \left(x - \frac{1}{2}\right)^2\right] \tag{5}$$

The slope of the surface of the airfoil can be solved to be:

$$Z' = 2\tau(1 - 2x) \tag{6}$$

The simulation will model the velocity potential across the surface of the airfoil and estimate the pressure distribution. The results will then be compared against empirical data [4] and pressure data from the *TSDE_19* MATLAB code to determine the simulations applicability to the subsonic and transonic flow regimes and analyze the limitations of the model. The simulation will be simplified by only analyzing half of the airfoil as at zero angle of attack the symmetric airfoil will have a symmetrical pressure distribution above and below the airfoil.

The empirical data [4] and the provided pressure coefficient data from the *TSDE_19* MATLAB code [2][3] utilize the transonic flow theory so utilizing the transonic flow parameter will allow for the comparison of data with the simulated results.

$$K = \frac{1 - M_\infty^2}{(M_\infty^2 \tau)^{\frac{2}{3}}} \tag{7}$$

III. Experimental Setup & Procedures

To compute a numerical solution of the Prandtl-Glauert equation, a discretized mesh of size *I* by *J* was created so that the potential at each point (i, j) could be calculated.

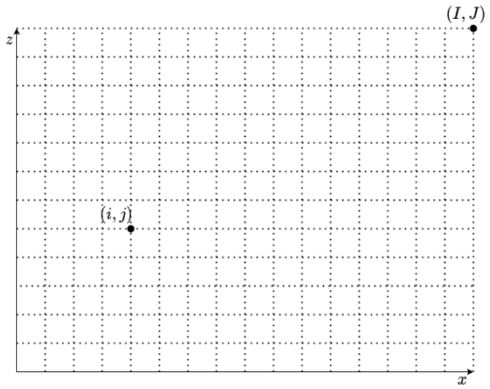


Fig. 3 Discretized mesh of size (I, J).

The boundary conditions of the left and right sides of the domain were approximated to be "far away" from the body. At this distance, the change in the potential is negligible and can be approximated as zero. If the mesh of the domain in Fig. 3 were to be extended in every direction by one unit to create a pseudo-mesh as seen in Fig. 4.

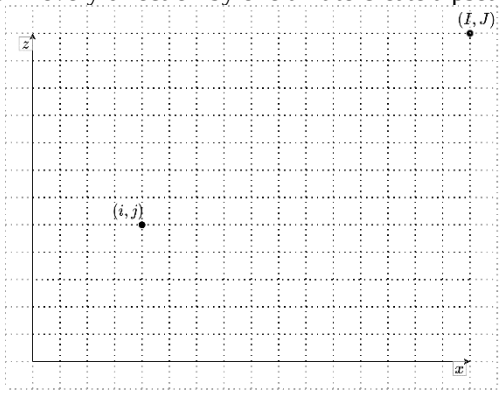


Fig. 4 Domain with extended pseudo-mesh.

With the pseudo-mesh in Fig. 4, the change in the potential of the vertical boundary on the left can be written as:

$$\frac{\partial \varphi}{\partial x} \approx \frac{\varphi_{2,j} - \varphi_{0,j}}{2\Delta x} = 0 \tag{8}$$

Where $\varphi_{0,j}$ represents the potential just outside of the left boundary. The right vertical boundary has the same condition as the left:

$$\frac{\partial \varphi}{\partial x} \approx \frac{\varphi_{I+1,j} - \varphi_{I-1,j}}{2\Delta x} = 0 \tag{9}$$

With $\varphi_{I+1,j}$ being the potential just outside the right boundary. This approximation can be applied to the upper boundary as is, but the lower boundary requires a special case as the potentials from the airfoil are present.

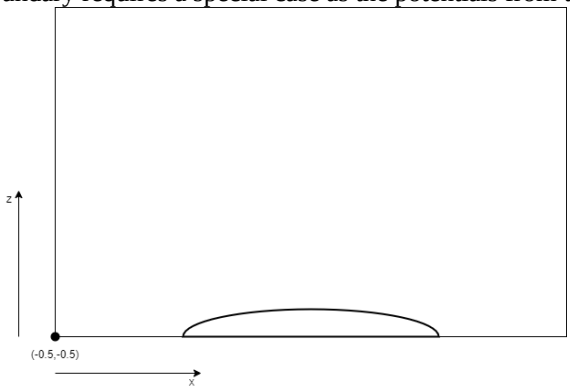


Fig. 5 Generalized layout of airfoil in the domain.

The disturbances from the airfoil are represented in the form of the slope of the airfoil from eq. 6, thus the parameter space can be visualized as seen in Fig. 6 of the parameter space. The domain was set so that the bottom left corner, or domain index $(1,1)$, was given the designation of $(-0.5, -0.5)$ and the upper right corner, domain index (I, J), was given the designation of $(1.5, 1.5)$ with a mesh resolution of 100 by 100. The potential of the airfoil was positioned at $z = -0.5$ and spanned $0 < x < 1$ as seen in Fig. 6 of the parameter space.

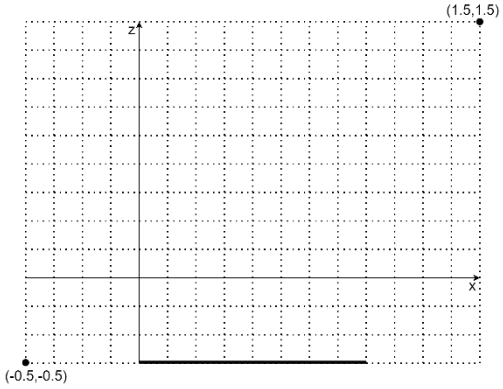
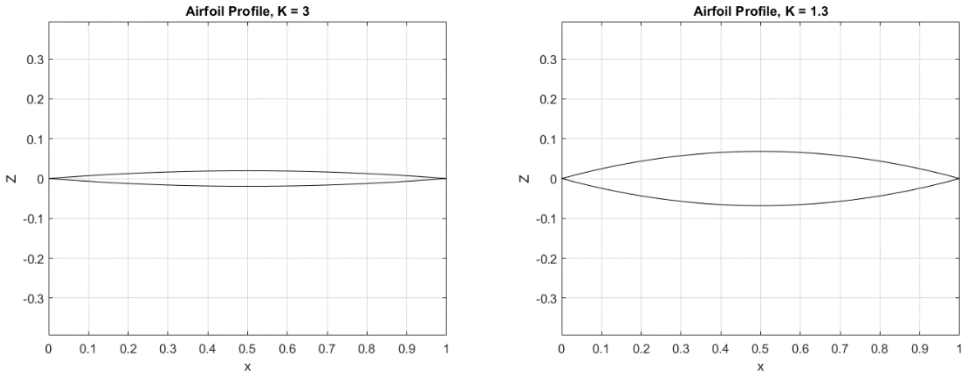


Fig. 6 Parameter space for simulation.

The parameter space changes due to the changing profile of the airfoil which is affected by the value of the free-stream Mach number and transonic similarity parameter and eq. 7.



(a) (b)

Fig. 7 Airfoil profiles for (a) $K = 3$ and (b) $K = 1.3$.

From the parameter space in Fig. 6 and the pseudo-mesh extension in Fig. 4 the change in the potential vertically across the lower boundary can be approximated as the piecewise function:

$$\frac{\partial \varphi}{\partial z} \approx \frac{\varphi_{i,2} - \varphi_{i,0}}{2\Delta z} = \begin{cases} Z'(x_i) & on\ body \\ 0 & off\ body \end{cases} \tag{10}$$

Solving for $\varphi_{i,0}$, the potential that lies one grid point outside of the boundary, gives:

$$\varphi_{i,0} = \begin{cases} \varphi_{i,2} - 2\Delta z Z'(x_i) & on\ body \\ \varphi_{i,2} & off\ body \end{cases} \tag{11}$$

Using the central difference approximation for the Prandtl-Glauert equation, eq. 4, gives the new equation that was used along with the boundary conditions in the simulation:

$$\beta^2 \frac{\varphi_{i-1,j} - 2\varphi_{i,j} + \varphi_{i+1,j}}{\Delta x^2} + \frac{\varphi_{i,j-1} - 2\varphi_{i,j} + \varphi_{i,j+1}}{\Delta z^2} = 0 \tag{12}$$

$\beta = 1 - M^2$ in eq. 12. The Gauss-Seidel Method for numerical approximations was used in the simulation and allows eq. 12 to be rewritten as:

$$\varphi_{i,j} = \left[\beta^2 \frac{\varphi_{i-1,j} + \tilde{\varphi}_{i+1,j}}{\Delta x^2} + \frac{\tilde{\varphi}_{i,j+1} + \varphi_{i,j-1}}{\Delta z^2}\right]\left[\frac{2\beta^2}{\Delta x^2} + \frac{2}{\Delta z^2}\right]^{-1} \tag{13}$$

The φ with the tilde represent the previous iteration's value for the potential while the φ without the tilde represent the current iteration. This allows for the solution to converge faster by using two updated values in every calculation. Once the entire simulation was completed, the pressure and drag coefficients were calculated using the following equations.

$$u_{i,j} = \frac{\varphi_{i+1,j} - \varphi_{i-1,j}}{2\Delta x} \tag{14}$$

$$c_p = -\frac{2u}{\tau^{\frac{2}{3}}} \tag{15}$$

$$c_d = 2\int c_p dZ \tag{16}$$

The horizontal velocity at every mesh point was calculated using eq. 14 which was then used in the calculation of the pressure coefficient in eq. 15. The drag coefficient was only calculated along the surface of the airfoil using the MATLAB function *trapz* for the integration of the pressure coefficient and is multiplied by 2 to get the drag for both the upper and lower surface of the airfoil.

There are some limitations with the method when operating in the transonic region. Because the Mach number used in the simulation is Mach 0.85 there is a high probability that supersonic flow appears at some point on the airfoil. In the case of supersonic flow, the flow is no longer isentropic especially when shock waves are present on the body of the airfoil. This simulation also does not work for airfoils at considerable angles of attack.

IV. Results

Both the subsonic and transonic cases were tested with a free-stream Mach number of 0.85. The transonic similarity parameter for the subsonic case was $K = 3$.

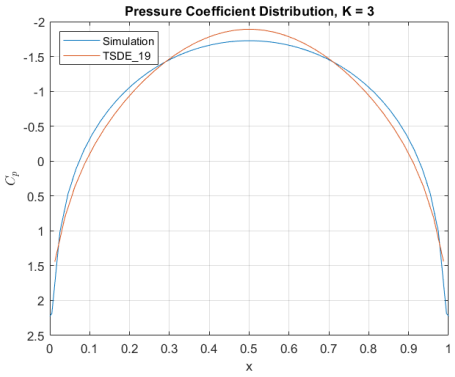


Fig. 8 c_p distribution along airfoil for $K = 3$ for simulation and *TSDE_19* data [3].

As seen in the subsonic regime, the simulated results line up closely with the provided data [3] and with the empirical data seen in Fig. 10a. The simulated results do not have the same range in pressure coefficient as the TSDE data, with the simulation results ranging from $-1.7258 < c_p < 2.1859$ while the TSDE data ranges from $-1.8895 < c_p < 1.4423$.

The transonic similarity parameter for the transonic case was $K = 1.3$.

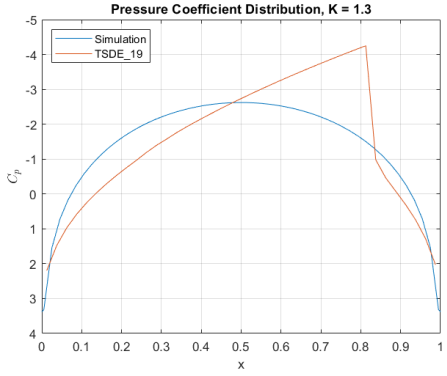


Fig. 9 c_p distribution along airfoil for $K = 1.3$ for simulation and *TSDE_19* data [2].

As seen in the transonic regime, the simulation is unable to predict the supersonic flow on the airfoil and cannot account for the shock jump around $x = 0.825$ seen in both the TSDE data [2] and the empirical data in Fig. 10b.

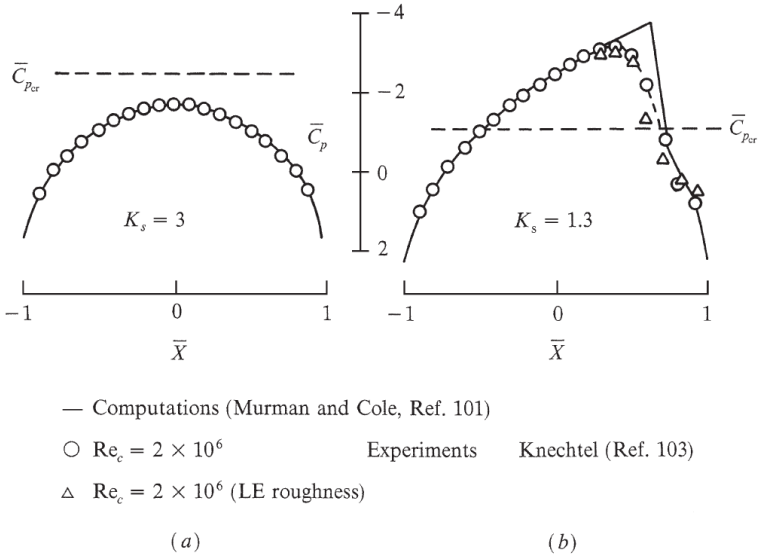


Fig. 10 Pressure coefficient distributions for a circular arc airfoil [4].

The calculated drag coefficient values were calculated using eq. 16 and recorded in Table. 1. It can be seen that the drag coefficients in the subsonic region have similar mantissas but the simulation is a factor of 10 less than the provided TSDE data [3]. While the difference between the magnitudes of the subsonic drag coefficient is significant between the two data sources, their magnitudes are both close to zero such that they could be considered negligible. This is not the case in the transonic case as the simulation's drag coefficient is closer to the TSDE c_d for the subsonic case. The transonic TSDE c_d cannot be considered negligible like the subsonic case where the simulation's transonic c_d could be.

Table. 1 c_d calculations from simulation and provided *TSDE_19* data [2][3].

Transonic Similarity Parameter, K	Simulation	*TSDE_19* Data
3	7.698272E-6	7.945606E-5
1.3	4.099671E-5	0.167581

V. Conclusions

The simulation proved to be accurate to the subsonic case compared to the empirical data [4] which was to be expected as subsonic assumptions were made to solve the Prandtl-Glauert equation. The discrepancy between the simulation's results and the TSDE and empirical data can be attributed to how the later utilize transonic assumptions when solving for their solution.

The large discrepancy between the simulation's results and the TSDE and empirical data for the transonic case was also expected due to the same reasons for the discrepancies in the subsonic case. The TSDE solutions are able to more correctly account for the transonic region as they can account for both local subsonic flow and local supersonic flow along the airfoil whereas the created simulation assumes that the flow across the whole airfoil is always subsonic.

It was expected that the drag coefficient for both the subsonic and transonic cases to be close to zero as for a thin airfoil at zero angle of attack as the oncoming air does not experience much resistance to flowing in a straight line. The difference between the transonic region results of the simulation and the TSDE data were also expected as in transonic flow a shock wave forms on the surface of the airfoil so the supersonic flow can return to subsonic. This shock wave introduces wave drag on the airfoil which is a major contributor to drag on transonic airfoils. The simulation did not account for the shock wave, so its drag coefficient was significantly less than the TSDE data.

The simulation results were able to show the reliability of the Prandtl-Glauert equation for purely subsonic flow and the application of numerical methods to small disturbances of complex airfoils. The simulation also shows the weaknesses of the Prandtl-Glauert equation at approximating transonic aerodynamics. The simulation could be improved in future experimentation by changing the simulation to accommodate for the transonic region and expanding the total testing domain to include both the upper and lower surfaces of the airfoil so that nonsymmetrical complex airfoil aerodynamics could be analyzed.

Appendix

A. Tabulated results

Table. 2 c_p data for $K = 3$ [3].

x Position	c_p	x Position	c_p
0.0125	1.4423	0.5125	-1.8895
0.0375	0.8114	0.5375	-1.8760
0.0625	0.3715	0.5625	-1.8489
0.0875	0.0359	0.5875	-1.8085
0.1125	-0.2378	0.6125	-1.7550
0.1375	-0.4710	0.6375	-1.6887
0.1625	-0.6748	0.6625	-1.6097
0.1875	-0.8558	0.6875	-1.5181
0.2125	-1.0179	0.7125	-1.4138
0.2375	-1.1637	0.7375	-1.2966
0.2625	-1.2947	0.7625	-1.1657
0.2875	-1.4121	0.7875	-1.0201
0.3125	-1.5166	0.8125	-0.8581
0.3375	-1.6083	0.8375	-0.6772
0.3625	-1.6875	0.8625	-0.4734
0.3875	-1.7540	0.8875	-0.2403
0.4125	-1.8077	0.9125	0.0334
0.4375	-1.8483	0.9375	0.3689
0.4625	-1.8756	0.9625	0.8087
0.4875	-1.8894	0.9875	1.4397

Table. 3 c_p data for $K = 1.3$ [2].

x Position	c_p	x Position	c_p
0.0125	2.1921	0.5125	-2.8047

0.0375	1.4754	0.5375	-2.9392
0.0625	0.9708	0.5625	-3.0706
0.0875	0.5786	0.5875	-3.1991
0.1125	0.2508	0.6125	-3.3249
0.1375	-0.0363	0.6375	-3.4481
0.1625	-0.2952	0.6625	-3.5689
0.1875	-0.5334	0.6875	-3.6875
0.2125	-0.7560	0.7125	-3.8041
0.2375	-0.9681	0.7375	-3.9185
0.2625	-1.1918	0.7625	-4.0311
0.2875	-1.3943	0.7875	-4.1416
0.3125	-1.5701	0.8125	-4.2502
0.3375	-1.7544	0.8375	-0.9791
0.3625	-1.9215	0.8625	-0.4632
0.3875	-2.0797	0.8875	-0.0795
0.4125	-2.2327	0.9125	0.3074
0.4375	-2.3814	0.9375	0.7423
0.4625	-2.5260	0.9625	1.2818
0.4875	-2.6671	0.9875	2.0313

B. Sample Calculations

Thickness of airfoil

$$K = \frac{1 - M_\infty^2}{(M_\infty^2 \tau)^{\frac{2}{3}}} \tag{7}$$

$$\tau = \left(\frac{1 - M_\infty^2}{M_\infty^{\frac{4}{3}} K} \right)^{3/2}$$

$$\tau = \left(\frac{1 - 0.89^2}{0.89^{\frac{4}{3}} * 3} \right)^{3/2}$$

$$\tau = 0.03894$$

Height of airfoil profile

$$Z = \pm 2\tau \left[\frac{1}{4} - \left(x - \frac{1}{2} \right)^2 \right] \tag{5}$$

$$Z = \pm 2 * 0.03894 * \left[\frac{1}{4} - \left(0.5 - \frac{1}{2} \right)^2 \right]$$

$$Z = 0.01947$$

Slope of airfoil profile

$$Z' = 2\tau(1 - 2x) \tag{6}$$

$$Z' = 2 * 0.03894 * (1 - 2 * 0.5)$$

$$Z' = 0$$

C. MATLAB Code

```
clc; clear;% close all

TSDEdata1_3 = importdata("TSDEdata1_3.txt").data;
TSDEdata3 = importdata("TSDEdata3.txt").data;

x = linspace(-0.5,1.5,100)'; z = linspace(-0.5,1.5,100)';
Nx = length(x); Nz = length(z);
I = Nx; J = Nz;
M = 0.85;
beta = sqrt(1-M^2);
K = 3;
```

```
k = 0;
t = (((1-M^2)/K)^(3/2))*(M^-2);
dx = (max(x)-min(x))/Nx; dz = dx;
g = 1.4;

for i = 1:Nx
   if x(i)>0 && x(i)<1
      z(i) = 2*t*(0.25-(x(i,1)-0.5)^2);
      zprime(i) = 2*t*(1-2*x(i));
   else
      z(i) = 0;
   end
end

maxerror = 1E-6;
error = 2*maxerror;
esum = zeros(I,J);

phi = zeros(I,J);
oldphi = zeros(I,J);

while error > maxerror
   if k > 10000
      break
   end
   k = k+1;
% boundary conditions
   for j = 1:J
      phi(j,1) = phi(j,3);
   end
   for i = 2:I
      for j = 1:J
         if i == I
            phi(j,i) = phi(j,i-2);
         elseif j == 1
            if x(i) < 0 || x(i) > 1
               phi(j,i) = phi(3,i);
            else
               phi(j,i) = phi(3,i) - 2*dz*zprime(i);
            end
         elseif j == J
            phi(j,i) = phi(j-2,i);
         else
            phi(j,i) = (beta^2*(phi(j,i-1)+oldphi(j,i+1))/dx^2+(phi(j-1,i)+oldphi(j+1,i))/dz^2)*(2*beta^2/dx^2+2/dx^2)^-1;
         end
      end
   end
   for i = 1:I
      for j = 1:J
         esum(i,j) = abs(phi(j,i)-oldphi(j,i));
         oldphi = phi;
      end
   end
end
for i = 2:I-1
   for j = 2:J-1
      u(j,i) = (phi(j,i+1)-phi(j,i-1))/2/dx;
      cp(j,i) = -2*u(j,i)/t^(2/3);
   end
end

z_theory_3 = 2*t*(0.25-(TSDEdata3(:,1)-0.5).^2);
```

```
c_d_3 = 2*trapz(z_theory_3,TSDEdata3(:,2))
z_theory_1_3 = 2*t*(0.25-(TSDEdata1_3(:,1)-0.5).^2);
c_d_1_3 = 2*trapz(z_theory_1_3,TSDEdata1_3(:,2))
cd = 2*trapz(z(26:75),cp(2,36:75))

figure
plot(x(1:Nx-1)',cp(2,:))
hold on; axis ij; grid on
plot(TSDEdata3(:,1),TSDEdata3(:,2))
xlim([0,1])
title("Pressure Coefficient Distribution, K = 3")
xlabel("x");ylabel("$C_{p}$","interpreter","latex")
legend("Simulation","TSDE\_19","Location","northwest")

figure
plot(x,z,'k')
hold on; grid on; axis equal
plot(x,-z,'k')
xlabel("x"); ylabel("Z");
title("Airfoil Profile, K = 3")
xlim([0,1])
```

Acknowledgments

S. M. Porembski would like to thank Dr. Valana Wells and Isaiah Wall for the introduction, explanation, and additional help with theory and information required for the completion of this experiment as it would not have been possible without their help.

References

[1] MATLAB, Matrix Laboratory, Software Package, Ver. R2023A, The MathWorks Inc., Natick, MA 2023.
[2] Wall, I., "TSDEdata1_3.txt," ASU Canvas, Available: https://canvas.asu.edu/courses/143786/files/65281954?module_item_id=10656744.
[3] Wall, I., "TSDEdata3.txt," ASU Canvas, Available: https://canvas.asu.edu/courses/143786/files/65281975?module_item_id=10656747.
[4] Anderson, J. D., "Modern Compressible Flow with Historical Perspective," 4th ed., McGraw Hill, New York, 2021.
[5] Talay, T. A., "Introduction to the Aerodynamics of Flight," NASA SP-367, 1975.