Transonic Flow for Symmetric and Supercritical Airfoils

Tatem Hedgepeth
Arizona State University Undergraduate Engineering Student, Tempe, AZ, 85282

The purpose of this experiment was to analyze the different flow properties over a NACA 0012 and a RAE 5214 airfoil from nearly incompressible flow to transonic flow. Numerical analysis was performed using the Gauss-Seidel finite difference method to evaluate the linearized transonic small perturbation potential equation for Mach numbers of 0.4, 0.6, 0.8, 0.85 and 0.9. Trends for C_P, C_D, C_L, M_{cr}, ϕ, and ϕ_x were calculated. It was observed that the location of the shock in transonic flow occurred closer to the trailing edge for the RAE 5214 airfoil and that the strength of the shock was weaker than for the NACA 0012 airfoil under the same conditions. These properties of the transonic flow lead to the 5214 producing less drag and maintaining aerodynamic stability and performance for higher free stream Mach numbers. Overall it was found that Gauss-Seidel's perturbation methods accurately depicted the flow properties that were expected.

Nomenclature

α	=	*Angle of Attack*
a	=	*Speed of Sound*
c	=	*Chord Length*
C_D	=	*Drag Coefficient*
C_L	=	*Lift Coefficient*
C_P	=	*Pressure Coefficient*
γ	=	*Ratio of Specific Heat of Gas*
M_∞	=	*Free Stream Mach Number*
M_{cr}	=	*Critical Mach Number*
ϕ	=	Perturbation Potential
ϕ_x	=	Velocity in the x-direction
P	=	*Pressure*
R	=	*Specific Gas Constant of Air*
T	=	*Temperature*
t/c	=	*Thickness to Chord Ratio*
U_∞	=	*Velocity*

I. Introduction

This experiment analyzes flow properties over a symmetric and supercritical airfoil in the transonic flow regime. Even though the free stream velocity may be subsonic, the flow is considered transonic when it reaches a local Mach number of one or greater at any point along its path. This is possible because the speed of the flow varies as it moves over the airfoil and can increase to higher values. Once the flow meets or exceeds a local Mach number of one along the airfoil, compressibility effects begin to come into play as well as the formation of local shock waves, typically on the upper surface of the airfoil. Although low Mach numbers may produce only weak shocks, once the Mach number is sufficiently high, strong shocks will occur which drastically and nearly instantaneously change the properties of the flow.

The formation of strong shocks can lead to a multitude of adverse effects such as wave drag, boundary layer separation and Mach tuck. Wave drag is a phenomenon where drag increases greatly with the introduction of strong shocks due to the energy lost and pressure differential across the shock. Boundary layer separation occurs aft of the shock due to the turbulent nature of the shock and the adverse pressure gradients produced. Mach tuck is experienced as the speed increases and the shock wave produced moves further aft, therefore shifting the center of pressure aft as well and producing a downward pitching moment.

Aircrafts designed to operate in transonic flow often utilize a supercritical airfoil, a type of airfoil specifically designed for better performance in transonic conditions. Supercritical airfoils typically have a leading edge with a larger radius followed by a flatter top surface, compensated for by a higher degree of camber in the back section of the airfoil. The flattened top surface causes a more gradual acceleration of the flow, weakening the shock produced and shifting its position further aft in order to minimize pressure losses.

For the purposes of this experiment, the flow will be analyzed over a two-dimensional airfoil while assuming steady, inviscid, small perturbations and transonic flow. These assumptions are used in order to allow for the simplification of the boundary conditions and the steady transonic small disturbance equations, at the cost of lower computational accuracy.

Euler's equations (Equations one through three) govern inviscid transonic flow.

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho V) = 0 \tag{1}$$

$$\rho \frac{DV}{Dt} = -\nabla \mathrm{p} \tag{2}$$

$$\rho \frac{Dh_0}{Dt} = \frac{\partial p}{\partial t} \tag{3}$$

To further reduce these equations, the velocity potential can be defined in Equation four when assuming an irrotational flow.

$$V = \nabla \varphi \tag{4}$$

Equation five combines the previous equations.

$$\left(1 - \frac{\varphi_x^2}{a^2}\right)\varphi_{xx} + \left(1 - \frac{\varphi_y^2}{a^2}\right)\varphi_{yy} + \left(1 - \frac{\varphi_z^2}{a^2}\right)\varphi_{zz} - \frac{2\varphi_x\varphi_y}{a^2}\varphi_{xy} - \frac{2\varphi_x\varphi_z}{a^2}\varphi_{xz} - \frac{2\varphi_y\varphi_z}{a^2}\varphi_{yz} = 0 \tag{5}$$

Next, small perturbations are assumed which yields Equation six where K (Equation seven) is the transonic similarity parameter.

$$[K - (\gamma + 1)\varphi_x]\varphi_{xx} + \varphi_{yy} + \varphi_{zz} = 0 \tag{6}$$

$$K = \frac{1 - M_\infty^2}{\left(\frac{t}{c}\right)^{\frac{2}{3}}} \tag{7}$$

The far field boundaries to the left and right and top of the airfoil are modeled by a doublet in order to simulate the effect of the airfoil interjected in to the flow. It is important to set the boundaries far enough away from the simulated airfoil so that the flow can be close enough to free stream properties at the boundaries. The bottom boundary is simulated as the physical surface of the airfoil such that its interaction with the flow can be modeled and numerically calculated.

II. Procedure

The airfoils chosen for evaluation in this lab were the NACA 0012 airfoil and the RAE 5214 supercritical airfoil. Data for the x and z-coordinates of the airfoils was obtained online and inserted in to the MATLAB code. Flow properties over each airfoil were tested for Mach number of .4, .6, .8, .85 and .9 and at an angle of attack of 0^o. These Mach numbers were chosen in order to show how the flow changes from nearly incompressible at the low end to transonic at the high end. They are also more weighted towards the transonic regime in order to give a more detailed look at the effects the airfoil shape has on the flow properties at these Mach numbers. For the NACA 0012 airfoil, only one surface was tested since it is symmetrical, where as the top and bottom surface of the RAE 5214 airfoil were tested separately. The shapes of these airfoils are shown in figures (1) and (2) below.

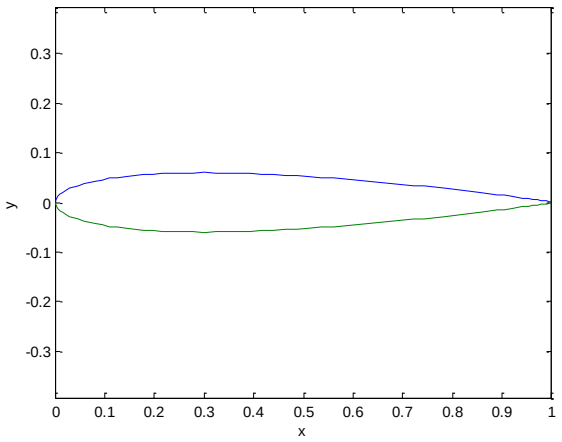

Figure 1. NACA 0012

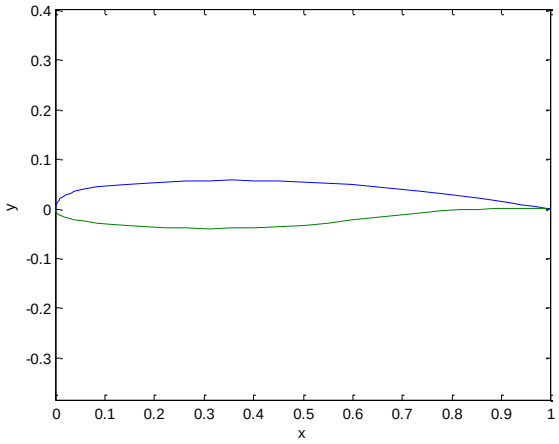

Figure 2. NACA RAE 5214

For this experiment, the Gauss-Seidel method was utilized to calculate flow properties. The Gauss-Seidel method is an iterative scheme which is based on a finite difference scheme. Each iteration creates a new solution based on the previous solution, with the goal of each new iteration converging towards more accurate results. Iterations were repeated until the results produced an associated error of less than 10% or 200 iterations were performed.

In order to apply this method, MATLAB was used to position the airfoil within a grid where the boundary conditions were applied at the edges. For the boundary conditions to remain valid, the edges of the grid needed to be placed far enough away from the airfoils. If the airfoil was placed too close to the edges of the grid, the flow would become supersonic at the edges and the boundary conditions would be violated. To avoid this issue for this experiment, the grid went from -6 to 6 in the x-direction and from 0 to 12 in the y-direction with a step size of 0.05 for each. In order to correctly position the airfoil the 'interp1' function was used to interpolate data points that correlate with the grid nodes for the x and z-coordinates of each airfoil using a spline method. The airfoil is positioned on the grid at -0.5 to 0.5 on the x axis.

In order to evaluate the flow over the airfoil, the linearized transonic small perturbation potential equation was used with its appropriate boundary conditions shown below in equations (8-10). In these boundary conditions, x_e and y_e are coordinates along the edges at $x_e = \pm L$ and $y_e = H$. The variables L and H represent the grid size, where L corresponds to the x-direction and H corresponds to the y direction.

$$\left[1 - M_\infty^2 - \frac{M_\infty^2}{U_\infty}(\gamma + 1)\emptyset_x\right]\emptyset_{xx} + \emptyset_{yy} = 0 \tag{8}$$

$$\emptyset_y(x, 0) = U_\infty \frac{df}{dx} \tag{9}$$

$$\emptyset(x_e, y_e) = \frac{D}{2\pi}\left(\frac{x_e}{x_e^2 + y_e^2}\right) \tag{10}$$

The boundary condition about the domain edges is a doublet with the strength D is obtained from the Murman and Cole paper[2]. Equation (11) represents this strength, where δ represents the airfoil thickness ratio, t/c. The free stream velocity of the air, U_∞, varies with each Mach number being evaluated.

$$D = \delta U_\infty \int f(x)dx + \frac{\gamma+1}{2}\frac{1}{\delta U_\infty}\iint u^2 dxdy \tag{11}$$

Using a matrix of zeros as an initialized value for ϕ, the top, right, and left boundary values were calculated. Then, the bottom boundary values were found based on the value of B, the far-field boundary shown in Eq. (12), which determines whether the flow is subsonic or supersonic. For a positive value of B the flow is subsonic and the discretized equation, Eq. (13), is used to evaluate ϕ. For a value of B less than or equal to 0, the flow is supersonic and Eq. (14) is used to evaluate ϕ.

$$\left[1 - M_\infty^2 - \frac{M_\infty^2}{U_\infty}(\gamma+1)\emptyset_x\right] \tag{12}$$

$$\phi_{i,j} = \frac{1}{2(B\Delta y^2 + \Delta x^2)}\left[B\Delta y^2\left(\phi_{i+1,j} + \phi^*_{i-1,j}\right) + \Delta x^2\left(2\phi_{i,j+1} - 2\Delta y \cdot U_\infty \cdot dfdx\right)\right] \tag{13}$$

$$\phi_{i,j} = \frac{1}{(B\Delta y^2 - 2\Delta x^2)}\left[B\Delta y^2\left(2\phi^*_{i-1,j} - \phi^*_{i-2,j}\right) - \Delta x^2\left(2\phi_{i,j+1} - 2\Delta y \cdot U_\infty \cdot dfdx\right)\right] \tag{14}$$

For the interior points of the grid, new values of ϕ are again calculated based on values of B. For subsonic flow, the interior points of the grid are found using Eq. (15) whereas for supersonic flow, ϕ is calculated from Eq. (16).

$$\phi_{i,j} = \frac{1}{2(B\Delta y^2 + \Delta x^2)}\left[B\Delta y^2\left(\phi_{i+1,j} + \phi^*_{i-1,j}\right) + \Delta x^2\left(\phi_{i,j+1} + \phi^*_{i,j-1}\right)\right] \tag{15}$$

$$\phi_{i,j} = \frac{\Delta x^2\Delta y^2}{(B\Delta y^2 + \Delta x^2)}\left[\frac{B}{\Delta x^2}\left(2\phi^*_{i-1,j} - \phi^*_{i-2,j}\right) + \frac{1}{\Delta y^2}\left(\phi_{i,j+1} + \phi^*_{i,j-1}\right)\right] \tag{16}$$

Equation (17) below shows the calculation of C_p, where u' is the change in velocity perturbation in the x-direction.

$$C_p = \frac{-2u'}{U_\infty} \tag{17}$$

In order to find u', the derivative of ϕ must be taken. By using the forward, backwards and central difference schemes, ϕ_x was calculated using equations (18-20) respectively.

$$\phi x(i.j) = \frac{-\phi(i+2,j) + 4\phi(i+1,j) - 3\phi(i,j)}{2\Delta x} \tag{18}$$

$$\phi x(i.j) = \frac{\phi(i-2,j) - 4\phi(i-1,j) + 3\phi(i,j)}{2\Delta x} \tag{19}$$

$$\phi x(i,j) = \frac{\phi(i+1,j) - \phi(i-1,j)}{2\Delta x} \tag{20}$$

Drag and lift coefficients for each airfoil surface were calculated using Eq. (21) and Eq. (22) below. For the RAE 5214 airfoil, drag values were added to find total drag while the lower surface lift value was subtracted from the top surface value to find total lift. For the NACA 0012 airfoil, the drag value was doubled to equal total drag and there was no lift due to symmetry. Cfx represents the force coefficient in the x-direction and cfy is the force coefficient in the y-direction. Eq. (23) and Eq. (24) are the coded forms of cfx and cfy, utilizing MATLAB's 'trapz' function which applies the trapezoidal rule in order to integrate about the surface of the airfoil.

$$C_D = cfx \cdot \cos(\alpha) + cfy \cdot \sin(\alpha) \tag{21}$$

$$C_L = cfy \cdot \cos(\alpha) - cfx \cdot \sin(\alpha) \tag{22}$$

$$cfx = trapz(\, f(y)\,, Cp\,) \tag{23}$$

$$cfy = trapz(\, f(x)\,, -Cp\,) \tag{24}$$

For each iteration of the code, values of B that were calculated as negative (indicating supersonic flow) were stored and later against the airfoil geometry in a case by case basis in order to see over what portions of the airfoil flow was supersonic.

The critical Mach number, M_{cr}, was estimated by running the code again for smaller Mach number increments preceding the case in which supersonic flow is first observed. In this way, the M_{cr} was calculated down to an uncertainty of .01.

III. Results

Presented below are plots for C_P, ϕ and ϕ_x values plotted vs. x with the airfoil surface located from -0.5 to 0.5.

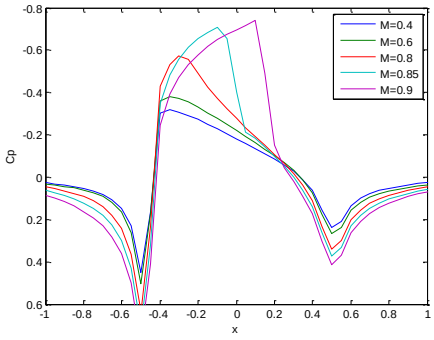

Figure 3. C_P, NACA 0012

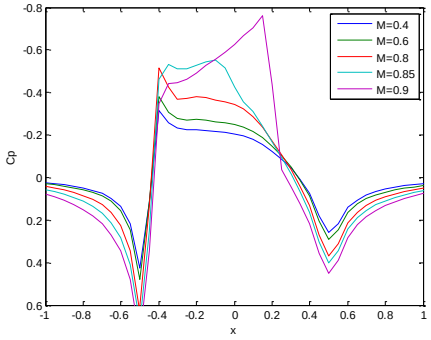

Figure 4. C_P, RAE 5214 Top

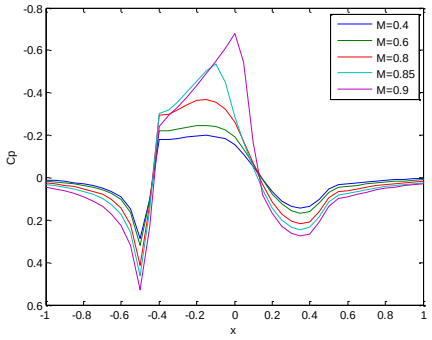

Figure 5. C_P, RAE 5214 Bottom

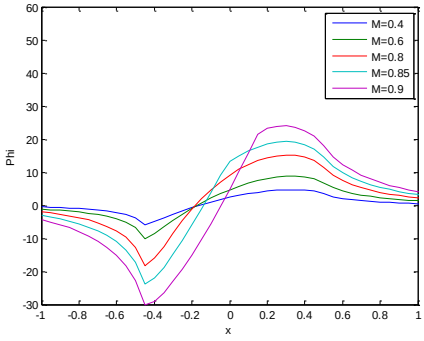

Figure 6. ϕ, NACA 0012

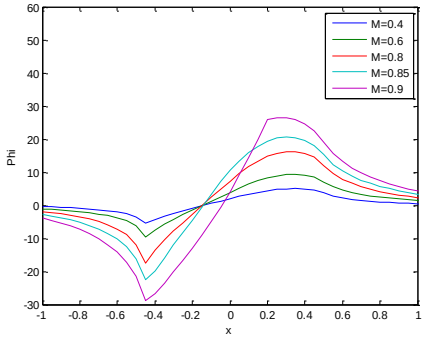

Figure 7. ϕ, RAE 5214 Top

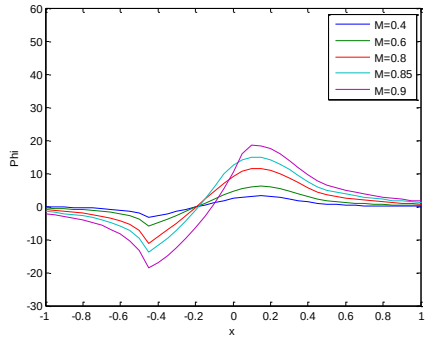

Figure 8. ϕ, RAE 5214 Bottom

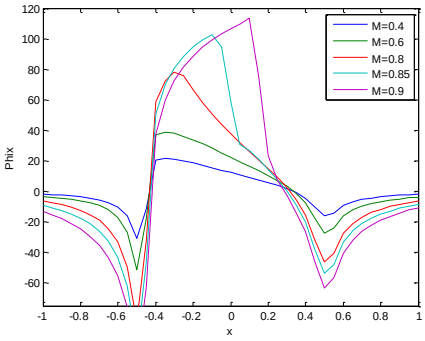

Figure 9. ϕ_x, NACA 0012

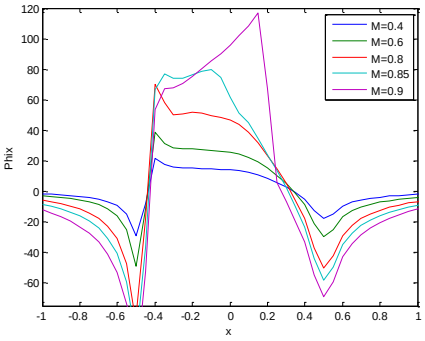

Figure 10. ϕ_x, RAE 5214 Top

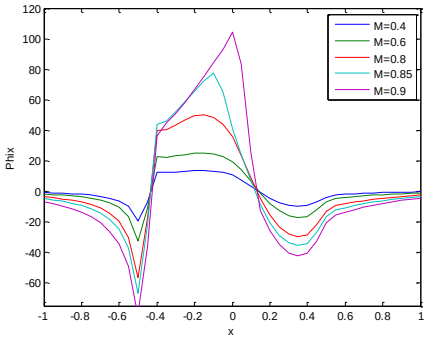

Figure 11. ϕ_x, RAE 5214 Bottom

For the NACA 0012 airfoil, it can be seen in the figure (3) that as Mach number increases, the location of lowest Cp value gets pushed further aft along the airfoil as well as increasing in magnitude before sharply dropping off. This can also be seen in figure (9) due to the peak velocity in the x direction increasing and moving further back. At Mach = 0.9, a large increase in pressure is observed as a strong shock forms around x = 0.1. The Cp trend starts changing the most from Mach = 0.8 and onward, which is due to the flow regime becoming transonic between M = 0.77 and M = 0.78, as seen in table (1). This trend also correlates with the regions of supersonic flow over the airfoil, which can be seen in figure (12) below. Once the Mach number reaches M = 0.8, the drag begins to increase greatly, indicating that it has surpassed the drag divergence Mach number and the wave drag losses will begin hampering aerodynamic performance.

	NACA 0012	Ra5214 Top	Ra5214 Bottom
$\mathbf{M_{cr}}$	.77-.78	.78-.79	.81-.82

Table 1. Critical Mach Number for Airfoil Surfaces.

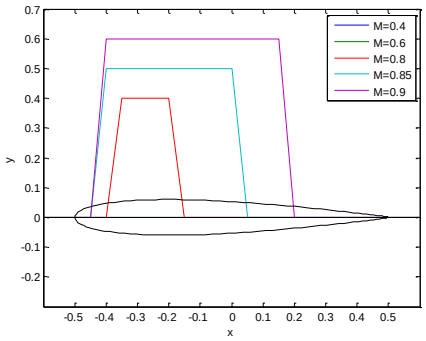

Figure 12. Supersonic Regions, NACA 0012

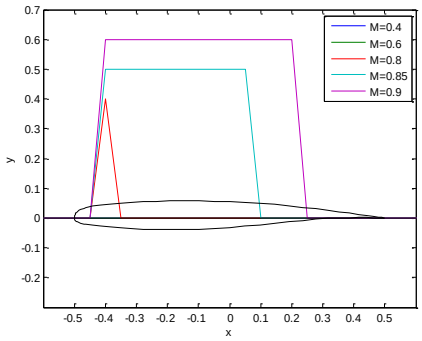

Figure 13. Supersonic Regions, RAE 5214 Top

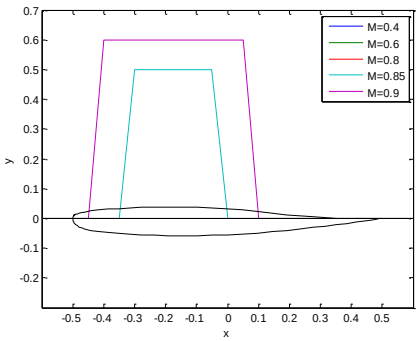

Figure 14. Supersonic Regions, RAE 5214 Bottom

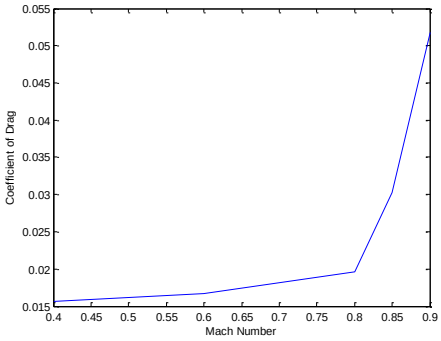

Figure 15. C_D, NACA 0012

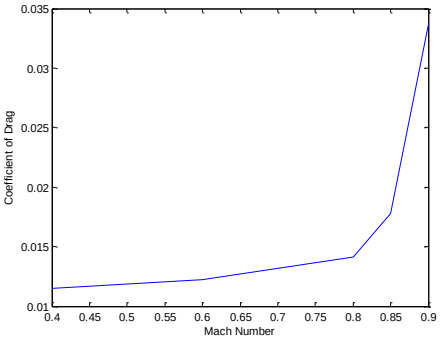

Figure 16. C_D, RAE 5214

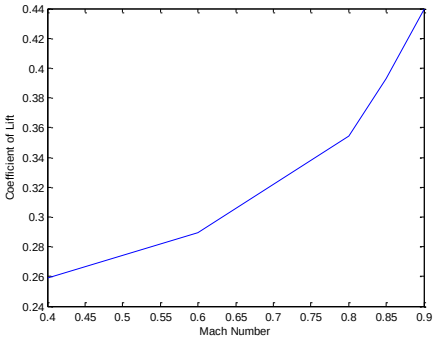

Figure 17. C_L, RAE 5214

The RAE 5214 airfoil shows marginal improvements over the NACA 0012 both in when it encounters the transonic regime and how it performs in transonic flow. The goal of the flattened top is to allow higher free stream Mach numbers before encountering transonic flow. Table (1) shows the slightly higher critical Mach number that the top surface has compared to the NACA 0012. This is further supported by examining figures (12) and (13). Whereas a significant portion of the 0012 airfoil is experiencing supersonic flow at M = 0.8, the top surface of the 5214 has only just breached the sonic barrier and the bottom surface is completely subsonic. It can also be seen in figures (4) and (10) that the flatter top surface of the 5214 produces a much gentler and smoother pressure distribution and velocity change up until the free stream Mach number reaches M = 0.9. Due to the airfoil's geometry, the shock wave produced by the flow is located further back than the shock wave on the symmetric airfoil. Figure (16) indicates that the drag divergence Mach number for the 5214 is larger than that of the 0012, and that the 5214's drag has a slightly smoother increase up until Mach numbers approaching M = 0.9. Additionally, as seen in table (2)

below, 5214's drag values are lower at every Mach number when compared to the 0012, which proves it has better aerodynamic performance as speed increases to the transonic regime.

M	C_D 0012	C_D RAE 5214	C_L RAE 5214
0.4	0.015687413	0.011486488	0.25871762
0.6	0.016698385	0.012231677	0.28934857
0.8	0.019655729	0.014140158	0.354206445
0.85	0.030235609	0.017804279	0.392700529
0.9	0.051831451	0.033800207	0.439843797

Table 2. C_D and C_L Values for Airfoils.

The goal of the RAE 52XX supercritical airfoil program was to create an airfoil which achieved a C_L of 0.4 at a Mach number as close to M = 0.8 as possible without drag-rise occuring[3]. From the numerical analysis of the 5214 airfoil, it seems to have gotten very close to achieving its goal. Although the C_L is a bit short of 0.4 at M = 0.8 (Figure (17)), its drag rise doesn't occur until around M = 0.85, at which point the C_L is nearly 0.4. Figure (18), published empirical results of the 5214 airfoil, shows how when Mach number increases, the region of supersonic flow increases further aft while the local Mach number remains fairly constant[3]. The numerical results from figure (11) at M = 0.8 also shows this smooth speed along the airfoil, a phenomenon which is not present in the 0012.

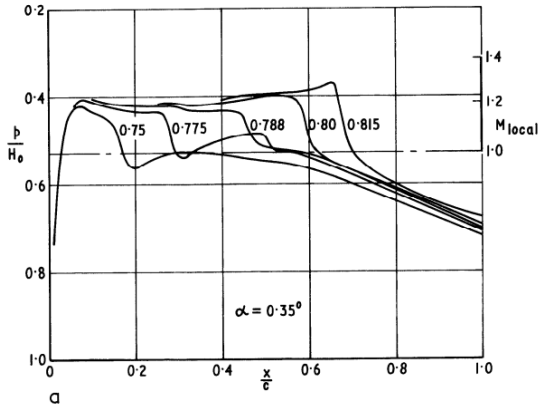

Figure 18. Supersonic Regions, RAE 5214

IV. Conclusion

Overall, it can be seen the supercritical geometry of the RAE 5214 airfoil performed better than the basic symmetric geometry of the NACA 0012 airfoil in the transonic regime. The main factors of this increase in performance lie in how the airfoils manipulate the supersonic portion of the flow that they encounter on their bodies. In comparison to the 0012, the 5214's flatter top allows a more gradual increase in speed, keeping the local Mach number stable and low enough to not produce a strong shock until the free stream Mach number increases even higher. The stronger shock produced by the 0012 creates larger adverse pressure gradients which lead to larger wave drag and boundary layer separation. Subsequently, the 5214 has lower drag, a higher critical Mach number, a higher drag divergence number, and more response for any control surface located on the trailing edge.

With respect to the Gauss-Seidel method, although the assumptions that were necessary were accompanied by a decrease in accuracy and robustness, the discretization of the velocity perturbation potential does give a good representation of how the flow is going to behave. In the industry, or in order to design a new or specifically tailored airfoil, this could be an efficient way to alter the parameters of an airfoil in order to see how aerodynamic performance is subsequently changed. A computational method of determining theoretical performance is much less expensive than producing a physical model of an airfoil and testing it in a wind tunnel. Additionally, with access to a

greater amount of computing power, software could be created to optimize a theoretical airfoil's geometry for given flight conditions.

Possible sources of error in the experiment are likely due to lack of computing power and the loss of accuracy due to the assumptions made. Although the results computed for the 0012 airfoil seemed indiscernible at a glance from those provided by Dr. Mertz, the code took minutes to run for each test, and most runs once the flow became transonic typically ended by hitting the max number of iterations rather than meeting the acceptable error. Any significant increase in accuracy would have been at the expense of minutes to hours of computing time. Additionally, the plots and figures were limited to only 20 points across the airfoil, due to the step size of 0.05. While this is generally acceptable, it lead to some figures having sharp edges and drop offs when in reality the curves should look smoother. Increasing the number of nodes in the grid and the number of iterations would likely bring results which show more detailed differences and nuances in the flows over the airfoils, but it was simply not feasible under the scope of this experiment.

Furthermore, the bottom surface of the RAE 5214 airfoil was inverted such that negative y values were positive, under the assumption that negative values would not allow the code to produce accurate results. This seemed to be the most logical set up at the time, due to the surfaces being calculated separately and the airfoil surface being the bottom boundary of the grid. However, since the slight hook at the trailing edge of the RAE 5214 airfoil actually goes above the chord line, a small portion of the lower surface coordinates go positive (or negative after they were inverted). This was a concept which was previously assumed impossible or inaccurate, but since the code ran seemingly flawlessly with it, it raises the question of whether or not the coordinates should have been inverted in the first place.

This experiment could be expanded by increasing the range of Mach numbers, i.e. seeing what would occur using the code for compressible flows ($M < 0.3$) or completely supersonic flows ($M > 1$). Varying the angle of attack would also be an interesting variable to study, if the computational method would allow for it. A final, more challenging addition would be to devise a way to run numerical analysis on both surfaces at once, with the airfoil in the middle of the grid (again, if the methodology would allow it).

References

1 Anderson, John D., Jr. *Modern Compressible Flow: With Historical Perspective*. Boston: McGraw Hill, 2003. Print.

2 Cole, J. D., and E. M. Murman. "Calculation of Plane Steady Transonic Flows." *AIAA Journal* 9.1 (1971): 114-21. Print.

3 Wilby, P.G. "The Design and Aerodynamic Characteristics of the RAE 5215 Aerofoil." *Aeronautical Research Council Current Papers* 1386 (1974). Web. 25 Apr. 2015. <http://naca.central.cranfield.ac.uk/reports/arc/cp/1386.pdf>.

Appendices

Appendix A. MatLab Code

```
clc;
clear all;
close all;
format short g;

% initializing airfoil data points

% NACA 0012:

x0t =
z0t =
x0b =

tc0 = .12;

% RAE 5214:

x5t =
z5t =

x5b =
z5b =
```

```
tc5 = .097;

%  plot(x0t,z0t,x0b,z0b)
%  axis equal
%  xlabel('x')
%  ylabel('y');
%
%  plot(x5t,z5t,x5b,-z5b)
%  axis equal
%  xlabel('x')
%  ylabel('y');

% initializing grid space
dx = .05;
dy = dx;
x = -6:dx:6;
y = 0:dy:12;

% adding airfoil geometry to grid space
query_points = (0:dx:1);
interp_x_points = interp1(x0t,x0t,query_points,'spline');
interp_y_points = interp1(x0t,z0t,query_points,'spline');

x_points = zeros(size(x));
y_points = zeros(size(y));
q = length(x);
p = length(y);

for i=1:21
    x_points(i+(q-1)/2-10) = interp_x_points(i);
    y_points(i+(q-1)/2-10) = interp_y_points(i);
end


% initializing matrices and vectors
phi_star = zeros(q,p);
phi = zeros(q,p);
phix = zeros(q,p);
u = zeros(q,p);
cp = zeros(size(phi_star(:,1)));
dfdx = zeros(1,q);
B = zeros(q,p);
supersonic_points = zeros(size(x));
black_line = supersonic_points;

yeT = max(y);
xeL = min(x);
xeR = max(x);

% calculating df/dx values
for i = ((q-1)/2)-10:1:((q-1)/2)+11;
    dfdx(i) = (y_points(i+1)-y_points(i-1))/(2*dx); % central
end

dfdx(((q-1)/2)-9) = (y_points(((q-1)/2)-9-2)-4*y_points(((q-1)/2)-9-1)+3*y_points(((q-1)/2)-9))/(2*dx); %
backward

dfdx(((q-1)/2)+12) = (-y_points(((q-1)/2)+12+2)+4*y_points(((q-1)/2)+12+1)-3*y_points(((q-1)/2)+12))/(2*dx); %
forward


% initializing constants
g = 1.4;
a = 340.29;

% initializing airfoil and Mach number choices
tc = tc5;
M = [.4 .6 .8 .85 .9];
cd = [0,0,0,0,0];
cl = [0,0,0,0,0];

M = [.76 .77 .78 .79 .80];


for r = 1:length(M)

    % initalizing error and iteration choices
    error = 2;
    error_allowed = .1;
    iterations = 1;
    maxiterations = 200;
```

```
    %calculating U_infinity
    Uinf = a*M(r);

    while error>error_allowed && iterations<maxiterations

         % calculating phix matrix
         for j=1:p
            phix(1,j)=(-phi(3,1)+4*phi(2,1)-3*phi(1,1))/(2*dx); % forward

            phix(q,j)=(phi(q-2,j)-4*phi(q-1,j)+3*phi(q,j))/(2*dx); % backward

            for i=2:q-1
                phix(i,j)=(phi(i+1,j)-phi(i-1,j))/(2*dx); % central
            end
         end


        % calculating doublet strength
        D = (tc*Uinf*trapz(y_points)*dx)+((g+1)/2)*((1/(tc*Uinf))*trapz(trapz(phix.^2)*dx)*dy);


        % calculating boundary conditions
        for j = 1:p;
            phi_star(1,j) = D/(2*pi)*(xeL/(xeL^2 + y(j)^2)); % left
            phi_star(q,j) = D/(2*pi)*(xeR/(xeR^2 + y(j)^2)); % right
        end

        for i=2:q-1
            phi_star(i,max(p))=(D/(2*pi))*x(i)/(x(i)^2+yeT^2); % top
        end


        % calculating B values
        for i = 1:q
            for j = 1:p
                B(i,j) = 1-M(r)^2-(M(r)^2/Uinf)*(g+1)*phix(i,j);
            end
        end

        % calculating bottom boundary
        for i=2:q-1
            if B(i,1)>0 % subsonic
                phi_star(i,1) = (B(i,1)/dx^2*(phi_star(i-1,1)+phi_star(i+1,1))+1/dy^2*(2*phi_star(i,1+1)-
2*dy*Uinf*dfdx(i)))/(2*B(i,1)/dx^2+2/dy^2);
            elseif B(i,1)<0 % supersonic
                phi_star(i,1) = (B(i,1)/dx^2*(-2*phi_star(i-1,1)+phi_star(i-2,1))+1/dy^2*(2*phi_star(i,1+1)-
2*dy*Uinf*dfdx(i)))/(-B(i,1)/dx^2+2/dy^2);

                %r
                % calculating supersonic region over airfoil
                %supersonic_points(i) = (r+1)/10;
            end
        end

        % calculating interior points
        for i=2:q-1
            for j=2:p-1
               if B(i,j)>0 % subsonic
                   phi_star(i,j) = ((B(i,j)/dx^2)*(phi_star(i+1,j)+phi_star(i-
1,j))+1/dy^2*(phi_star(i,j+1)+phi_star(i,j-1)))/(2*B(i,j)/dx^2+2/dy^2);
               else   % supersonic
                   phi_star(i,j) = (B(i,j)/dx^2*(-2*phi_star(i-1,j)+phi_star(i-
2,j))+1/dy^2*(phi_star(i,j+1)+phi_star(i,j-1)))/(-B(i,j)/dx^2+2/dy^2);
               end
            end
        end

        error = norm(abs(phi_star-phi));
        iterations = iterations + 1;
        phi = phi_star;

    end

    % calculating cp
    for i=1:q
         cp(i)=-(2*phix(i,1))./Uinf;
    end


%     % calculating lift and drag
```

```
%     alpha = 0;
%     cfy=trapz(x_points,-cp);
%     cfx=trapz(y_points,cp);
%     cd(r) = cfx*cos(alpha)+cfy*sin(alpha);
%     cl(r) = cfy*cos(alpha)+cfx*sin(alpha);
%
%
%     figure(1)
%     plot(x,cp);
%     axis ij
%     axis([-1 1 -.8 .6]);
%     if r == 5;
%         xlabel('x')
%         ylabel('Cp')
%         legend('M=0.4','M=0.6','M=0.8','M=0.85','M=0.9')
%     else
%     end
%     hold all
%
%     figure(2)
%     plot(x,phi_star(:,1))
%     axis([-1 1 -30 60]);
%     if r == 5;
%         legend('M=0.4','M=0.6','M=0.8','M=0.85','M=0.9')
%         xlabel('x')
%         ylabel('Phi')
%     else
%     end
%     hold all
%
%     figure(3)
%     plot(x,phix(:,1))
%     axis([-1 1 -75 120]);
%     if r == 5;
%         legend('M=0.4','M=0.6','M=0.8','M=0.85','M=0.9')
%         xlabel('x')
%         ylabel('Phix')
%     else
%     end
%     hold all

%     figure(4)
%     plot(x,supersonic_points(:))
%     axis([-.6 .6 -.3 .7]);
%     if r == 5;
%         legend('M=0.4','M=0.6','M=0.8','M=0.85','M=0.9')
%         xlabel('x')
%         ylabel('y');
%         plot(x0t-.5,z0t,'k',x0b-.5,z0b,'k',x,black_line,'k')
%     else
%     end
%     hold all


    iterations_this_run = iterations
end

%  figure (4)
%  plot(M,cd)
%  xlabel('Mach Number')
%  ylabel('Coefficient of Drag');
```